HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Automation and Systems Technology

Melak Mekonen Zebenay

Manipulator Control for Physical Astronaut-Robot Interaction

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology

Espoo August 11, 2009

Supervisors:

Professor Aarne Halme
Helsinki University of Technology

Professor Kalevi Hyyppä
Luleå University of Technology

Instructor:

Seppo Heikkilä
Helsinki University of Technology

Acknowledgment

I wish to express sincere appreciation to my instructor Seppo Heikkilä for his guidance and assistance throughout this thesis work, to Prof. Aarne Halme, Prof. Kalevi Hyyppä and Mr.Tomi Ylikorpi for their valuable consultation, to Mr.William Martin for his valuable language support.

Many thanks to all people in the Automation department and SpaceMaster round III students for making my experience here very enjoyable as well as all their support on various occasions, in various capacities and to various extents especially, Sami Kielosto and Miki.

I take this opportunity to record my indebtedness to the Erasmus Mundus that sponsored my Master's study.

Finally, I humbly thank my family for blessing me with abundant love and support all the time. My distance from home during this study and what I missed as a result, helped me realize just how large and precious that part in my life really is.

Espoo, August 11, 2009

Melak Mekonen Zebenay

I dedicated this thesis to my parents
and for my late uncle Sheferaw Zebenay.

Helsinki University of Technology Abstract

Author:	Melak Mekonen Zebenay
Title of Thesis:	Manipulator Control for Physical Astronaut-Robot Interaction
Date:	11. elokuuta 2009 **Number of pages:** 75
Faculty:	Faculty of Electronics, Communications and Automation
Department:	Automation and Systems Technology
Program:	Master's Degree Programme in Space Science and Technology
Professorship:	Automation Technology (Aut-84)
Supervisors:	Professor Aarne Halme (TKK)
	Professor Kalevi Hyyppä (LTU)
Instructor:	Seppo Heikkilä

The future of human-robot interaction applications is highly dependent on the capability to perform safe and efficient physical interaction using the robot's manipulators with a human. Even though the optimal use of Human-Robot Interaction (HRI) is a challenging problem, it can improve the quality and efficiency of the task execution since humans and robots are both able to perform certain tasks in some situations better than the others.

WorkPartner, TKK's mobile service robot, has no support for manipulator control system for the purpose of safe human-robot physical interaction application. Additionally, the existing manipulator control system has no position controller to be upgraded for human-robot physical interaction applications. This thesis addresses those problems by developing the compliance control capabilities of the *WorkPartner* manipulator. First, the state-of-the-art of physical human robot interaction is presented focusing on commonly used manipulator control algorithms, such as stiffness control and impedance control as well as force/torque sensors, such as the six-axis force/torque sensor and motor current sensor. Second, a manipulator control algorithm is suggested based on admittance control. This algorithm is implemented on the *WorkPartner* simulator and on the *WorkPartner* manipulator to examine the four selected manipulator behavior modes which are follow movement, hold position, adapt movement and push with force. The thesis test results show that selected admittance based manipulator control algorithm is capable to provide all the four examined manipulator behavior modes.

The thesis is part of the SpacePartner project which is a co-sponsored PhD project of the European Space Agency (ESA) and the Helsinki University of Technology (TKK).

Keywords: Impedance Control, Admittance Control, position control.

Contents

List of Figures

Symbols and Abbreviations

$\mathcal{A}_{k,i}[\beta]$	angle difference function
$\mathcal{B}_{n,m}^{k,i}$	matrix element correlation function
a_{Ti}	i^{th} linear tangent space position controller
q	joint variable
$\dot{q}$	joint velocity
$\ddot{q}$	acceleration
$M(q)$	the inertia matrix of the robot
$C(q, \dot{q})$	Coriolis and centripetal forces
$G(q)$	gravity forces
τ_d	disturbance torque
τ	input torque
$F(q)$	friction force
$N(q, \dot{q})$	nonlinear term
x_{Tdi}	i^{th} desired position trajectory tangent to the environment surface
k_{Tvi}	and K_{Tpi} are being the ith positive control gains
$\tilde{x}_{Ti}$	tangent space position tracking error
a_{Nj}	jth linear normal space force controller
k_{ej}	jth component of environmental stiffness
f_{Ndj}	jth component of the desired force exerted normal to the environment
K_{Nvj}	being the jth positive control gains
K_{Npj}	being the jth positive control gains
f_{Nj}	jth component of the force exerted normal to the environment

CAN	Controller Area Network
DLS	Damped Least Squares
DLR	German Aerospace Center
ESA	European Space Agency
HRI	Human-Robot Interaction
NASA	National Aeronautics and Space Administration
ODE	Open Dynamics Engine
PD	Proportional Derivative
PHRI	Physical Human Robot Interaction
PID	Proportional Integrator Derivative
SVD	Single Value Decomposition
TKK	Helsinki University of Technology

Chapter 1

Introduction

The intensification of the exploration of space increases the number and nature of space missions such as lunar and Mars missions. In the future, a cooperation between humans and robots becomes a vital technology for the success of challenging space missions. HRI can improve the quality and efficiency of the tasks as humans and robots perform differently on different tasks. The report done by (Fong and Nourbakhsh, 2004) shows that humans have better performance with cognitive and perception sensing whereas robots perform better with reactive, precise, and physically demanding functions. Depending on the kind of missions, some require relatively accurate performance of cognitive, and perception functions, while on the other hand, other tasks demand reactive, precise, and physically demanding functions. Even if the optimal use of HRI is a challenging problem, for future space exploration we need effective, efficient and natural HRI (Fong and Nourbakhsh, 2004).

Human-robot cooperation can enable exploration of the Moon, Mars, and even large-scale construction in extraterrestrial places. HRI development will be a necessity for these future missions which has profound advantage, such as human crews will be lesser in number, astronauts will therefore need robot assistance, there will be less costs to send robots rather than astronauts, minimizing the risk because robots are less sensitive to radiation compared to humans (Bluck and John, 2005).

These days, there has been increasing amount of research done for astronaut-robot cooperation development. Figure 1.1 outlines the vision of cooperation astronaut-*WorkPartner* artistic impression of *WorkPartner*. Similarly, Figure 1.2 shows the test done on ground by NASA to demonstrate the vision of robonaut-astronaut cooperation. A NASA Ames has also for example tested on ground collaborative control, by which humans and robots communicate with each other and work as partners. Furthermore, they even plan to build robots that have similar reasoning mechanisms to human beings (Bluck and John, 2005) which could make cooperation work suitable and safe.

WorkPartner is service robot that has rich features such as multi modal-interface and hybrid locomotion system which is suitable for space applications. However the *WorkPartner* manipulator control system system has no support to use it for astronaut-robot physical interaction applications since *WorkPartner* has only unaccessible position control system to be upgraded for Physical Human Robot Interaction (PHRI) applications. This thesis addresses those problems by selecting appropriate sensors and control algorithm for safe and efficient PHRI. The thesis approach is to examine the problem of PHRI through examining four well defined demonstration cases in mind, the findings from the demonstration cases can be applied to the problems of *WorkPartner* manipulator control for PHRI.

1.1 Thesis Goals

The goal of this thesis is to develop the capability of *WorkPartner*, also referred to as SpacePartner, to perform safe and efficient physical interaction using its manipulators with the human. In other words *WorkPartner* is required to respond intentionally to a force applied on the end effector which needs to have compliance control based on the applied force on the end effector. The goals of this thesis are summarized as follows :

- to analyze the manipulator performance requirement for cooperation purposes.

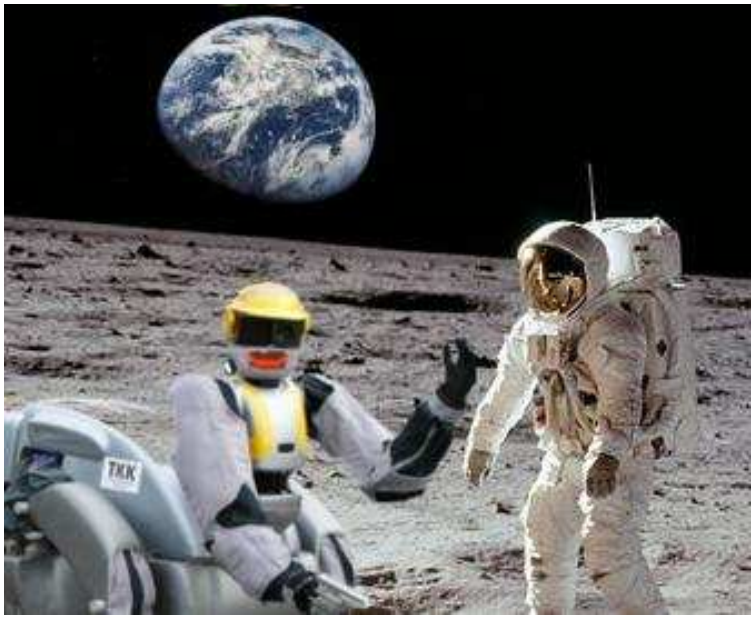

Figure 1.1: WorkPartner cooperating with an astronaut, artistic illustration (Heikkilä, 2008)

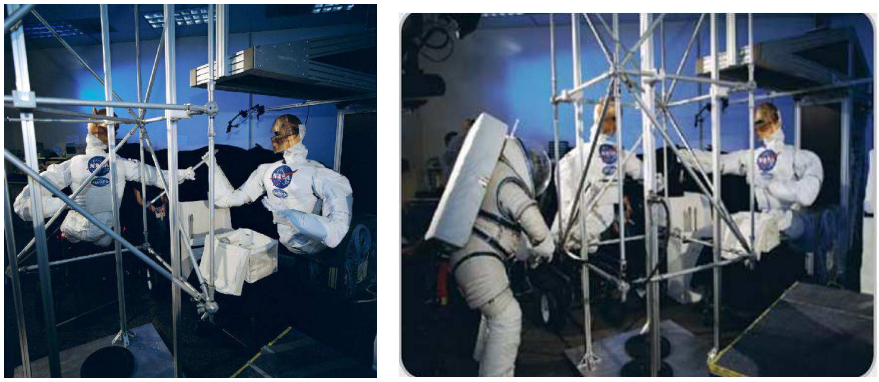

Figure 1.2: Robonaut provides hands-on assistance in space (NASAAmes, 2005)

- to identify and implement the appropriate sensors and/or controllers on the arm of the *WorkPartner*.
- to research the possible control algorithms that could be used for compliance control.
- to modify the SimPartner (Heiskanen et al., 2008), WorkPartner simulator, to suit the requirement of this thesis.
- to define and test the chosen algorithm on the modified *WorkPartner* simulator and on the real WorkPartner robot.

1.2 Thesis Outline

This thesis is organized in to six chapters as follows:

- Chapter 2 presents a survey of state-of-the-art in robot manipulator control algorithms and force/torque sensors for cooperation purposes. In addition chapter addresses PHRI to manipulate objects, specifically control of robot manipulator for cooperation work.
- Chapter 3 presents the *WorkPartner* manipulator control design and modeling. It describes the direct kinematics and inverse kinematics model of the *WorkPartner*. In addition, the chapter presents the control algorithms that are implemented in the SimPartner simulator.
- Chapter 4 presents the astronaut-robot interaction demonstration cases that are examined in the thesis and the hardware implementation of the algorithm on the *WorkPartner*. The selected demonstration cases are studied in detail. In addition this chapter presents analysis of the results from the four PHRI demonstration cases, presented in Chapter 3.
- The last chapter, Chapter 5, summarizes and presents conclusions of the thesis.

Chapter 2

Literature Review

"One of the symptoms of an approaching nervous breakdown is the belief that one's work is terribly important." Bertrand Russell

Many works related to human-robot assistance are currently underway or have been in development (Albu-Schaffer et al., 2008), (Diffler et al., 2003), (Kosuge and Kazamura, 1997),(Tsumugiwa et al., 2002). The algorithms and sensors used in the development of human-robot assistance are reviewed here. Section 2.1 presents the state-of-the-art of cooperation between human and robots, Section 2.2 presents an overview and comparison of commonly used force sensors for robot manipulators. Finally, Section 2.3 presents a computed-torque control algorithm as well as commonly used robot manipulator control algorithms for the purpose of human-robot cooperation.

2.1 PHRI in object manipulation

Nowadays, PHRI becomes an important research area where significant results are coming to the space missions and market such as *Robonaut* from the National Aeronautics and Space Administration (NASA), and *DLR-LWR-III* from the German Aerospace Center (DLR). However, there are still many interesting

challenges yet to be solved. A detailed report about requirement assessment, identifying the operational domain and other specification for robot-astronaut assistants have been done by (Cabrol et al., 1999) and (Heikkilä et al., 2008). This part of the literature review focuses on state-of-the-art robot manipulator control to perform tasks in cooperation with humans - which is one requirement of HRI in particular for future Mars or Moon missions.

The control of manipulators for industrial robots has been well researched and these robots are required to be precise, fast and as efficient as they are now. Typically, the control of the industrial robots is implemented with position control since the environment is well known (De Santis et al., 2008). However when it comes to the cooperation of humans and robots in unknown environments, position control will not be enough.

The paper (Albu-Schaffer et al., 2008) presents the methods for facilitating the safe interaction to robots with humans in unknown environments. That is, it presents two approaches to have safe HRI which are first, torque controlled and second , variable compliance actuation which is mainly in hardware. This paper (Albu-Schaffer et al., 2008) also recommends a mechanical design approach to achieve the required lightweight robots with the desired performance. Among the proposed mechanical criteria which make it easier to "implement compliant behavior and a smooth, vibration-free motion" (Albu-Schaffer et al., 2008) as well as most importantly safer interaction, are full-state measurements in the joints and sensor redundancy. Based on the mechanical design approach, the DLR has designed the lightweight robot arm shown in Figure 2.1.

(Albu-Schaffer et al., 2008) also illustrate the implementation of the compliant control law which reduces the effects of the joint friction and dampens the vibrations related to the joint compliance resulting from the torque signal. In addition, the motor position feedback produces the desired compliant behavior.

The compliant actuation can negatively affect the performance in terms of increased oscillations and settling times. However they are useful as a protection against unexpected contacts during PHRI (Siciliano and Villani, 2007).

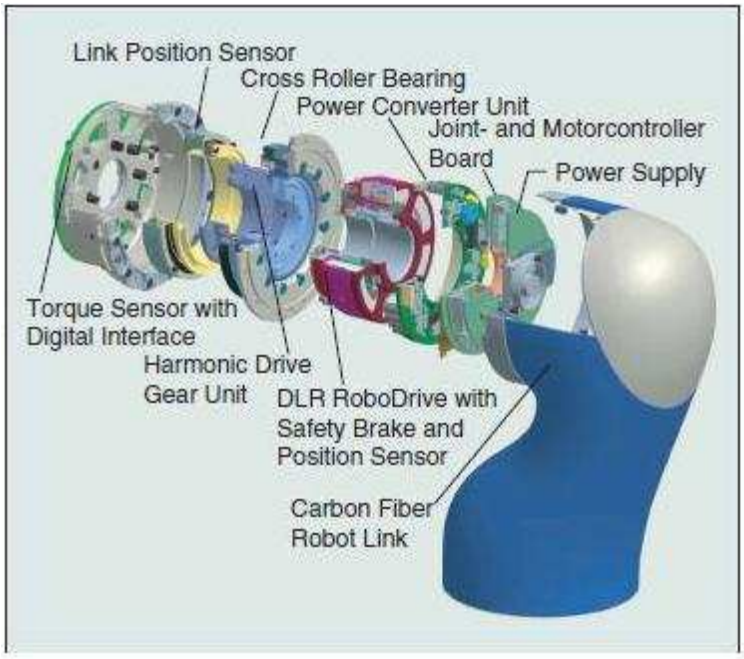

Figure 2.1: The mechatronic joint design of the DLR-LWR-III (Albu-Schaffer et al., 2008)

Unlike DLR, (Siciliano and Villani, 2007) suggest vision and force based control for physical interaction which could lead to improved performance, without necessarily considering the mechanical design. The authors also noticed that if the vision sensor is used for controlling force in addition to the contact sensor, it will improve the stability margins. Moreover, the resulting system has the capacity of approaching a rigid surface with a higher speed and less impact force, likewise the paper from (Nelson et al., 1995) affirms this conclusion. One similar work by (Morel et al., 1998) highlights the growing interest of vision based control. The use of only a vision sensor may not be enough due to changing environmental conditions which shows that sensor fusion is important. Unfortunately, there has been little work done on achieving the sensor fusion of contact sensors and visual sensors (De Santis et al., 2008).

Robonaut, the humanoid robot designed by NASA (Diffler et al., 2003), use vision as a key component for interacting with objects, monitoring human motion and estimating the tool pose. Furthermore robonaut implemented its control system with the combination of tele-operation, shared control and autonomy

which forms a distributed model. The control system of Robonaut illustrates the usefulness of Compliance Control for the application of PHRI as in the paper (Albu-Schaffer et al., 2008).

The work of (Kosuge and Kazamura, 1997) demonstrates cooperation work between a human and a robot. The paper illustrates the motion generation from the applied force on the end-effector of a robot manipulator through object that is handled by the robot. The motion is proportional to the applied force from the human. Furthermore, (Kosuge and Kazamura, 1997) suggested four types of control algorithms to generate the proportional motion from the applied force which are:

- Force augmentation
- Position Control type-stiffness
- Velocity Control type-damping
- Acceleration Control type-impedance

The authors implemented two of the methods and illustrated how human-robot cooperation can handle an object in using impedance and damping control.

2.2 Force Sensing Techniques for Manipulator Control

Force/torque sensing techniques are reviewed in this section to identify suitable sensor for human-robot interaction. To make this interaction safe, efficient and able to reach a good level of performance, the necessary sensors have to be developed and chosen wisely. The sensor information should be able to reach the expected performance. This increases the safety during interaction with humans or in unknown environments. The most common force sensing techniques that have been used for controlling robot manipulators are reviewed here.

2.2.1 Six-Axis Force/Torque Sensors

The six-axis force/torque sensors, typical example shown in Figure 2.2, measures all six components of force and torque which are three components of force and three components of torque. These sensors are the most complex of all the sensors covered in this literature review. The working principle of these sensors depends on the strain gauges working principle which are comprised of six silicon strain gauges.

Such sensors can be used to implement control algorithms, such as active stiffness control, hybrid position/force control, and impedance control. However, these type of sensors cannot detect collision between the manipulator and the environment. Furthermore they are risky to use for space mission without modification or without proper thermal control since high temperature changes affect the output signal of the strain gauges. Typically, six-axis force/torque sensors are implemented between the manipulator arm and the end-effector to measure and feedback the forces and torques from the manipulated object (Sum, 2008). The analysis done by the NASA Jet Propulsion Laboratory (Okon and DiCicco, 2005) shows the following characteristics of the sensors. The accuracy is quite good relative to the others and repeatability is satisfactory.

2.2.2 Joint Torque Sensor

Joint Torque Sensor sensors measure the load in joint space as a joint torque, unlike the six-axis force/torque sensors that transform the actual load to Cartesian space directly. Using the Jacobian equation 2.8 that relates torque and force, it is possible to transform the force (load) data in the end effector of the manipulator. There sensing technique uses strain gauges with specific configuration. The report from the NASA Jet Propulsion Laboratory (Okon and DiCicco, 2005) showed that these sensors are less accurate than six-axis force/torque sensors. This analysis report also shows that the dominant sources of error are a disturbance torques as well as false torque measurements. Furthermore, the repeatability of the sensors is affected by the weight of the manipulator and

Figure 2.2: ATI multi-axis force/torque sensor, (Sum, 2008)

the effectiveness of the sensors are highly dependent on the position of the manipulator. Unlike six-axis force/torque sensors, these sensors are able to detect collisions of the arm with the environment. In addition these sensors cannot measure the moments at the contact point and are unable to estimate the contact point location.

Similar to force/torque sensors, these sensors have a risk for use in space missions or environments where there are variations in temperatures due to there sensing technique. The sensing principle use strain gauges with different configuration to six axis force/torque sensors. The variations in temperature will affect the accuracy, repeatability and other will cause unexpected effects on the strain gauge reading.

2.2.3 Link Strain Gauge Sensors

These sensors, example shown in Figure 2.3, can measure torque after calibration with the sensor direct output voltage. In theory, the moment or torque is the cross product of a vector from the contact point to the location of the sensed moment as of Equation 2.1. Unfortunately, the cross product is not invertible, which makes the computation of force on the end effector of the manipulator

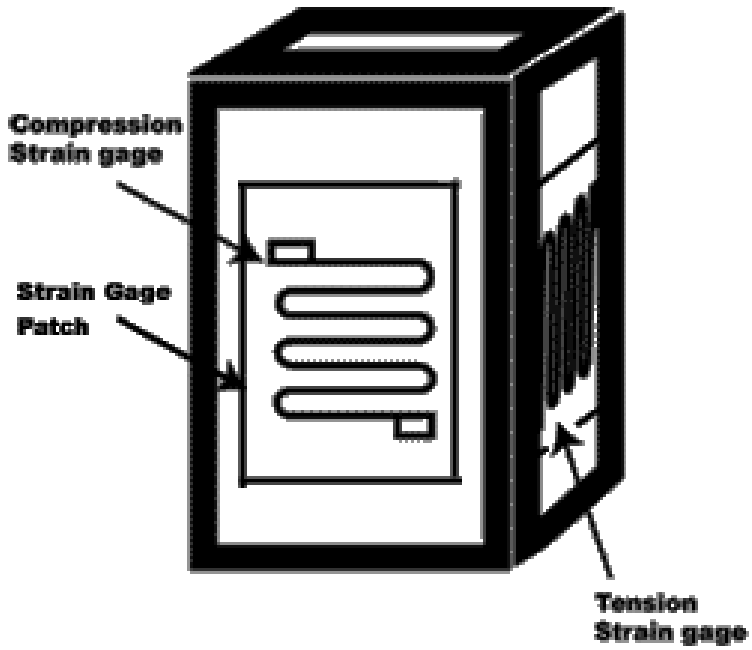

Figure 2.3: 2D version of strain gauge, (SOF, 2009)

overdetermined. Nevertheless, the best estimation of force can be determined using the least square approach from Equation 2.1 (Okon and DiCicco, 2005). The analysis report by NASA's Jet Propulsion Laboratory suggests that the accuracy of these sensors are not good due to likely error both mechanical and in force estimation, and has the same repeatability as joint torque sensors. These sensors can detect collision but cannot measure the moments at the contact point (Okon and DiCicco, 2005). The equation that relate force and torque is given below.

$$\vec{\tau} = \vec{r} X \vec{F} \tag{2.1}$$

where $\vec{\tau}$ is sensed torque, $\vec{r}$ is the vector from the contact point to the location of the sensed torque, and $\vec{F}$ is the force at the end effector.

2.2.4 Motor Current Sensors

The motor current sensors consist of a resistor coupled with the current mirror. Figure 2.4 shows motor current sensor in the product LMD18200 H-bridge. As the name implies the motor current sensors give the current measurement so that the joint torques can be estimated using the motor's corresponding torque constant, gear ratio, and gear train efficiency. After estimating the torque, it is possible to calculate the force components using the Jacobian Equation 2.8. The investigation of the Jet Propulsion Laboratory (Okon and DiCicco, 2005) shows these sensors have good repeatability, have good accuracy in the direction of applied force, and also they are accurate when the force is being actively applied by the arm. However after some load threshold value the motor currents become uncorrelated to the applied load which is a problem unique to these sensors according to the same report. These sensors have been used in space and it has a well specified flight design requirement (Okon and DiCicco, 2005).

2.3 Robot Manipulator Control Algorithms

This section reviews the commonly used algorithm for HRI application as well as to control force/torque. The chosen algorithms are presented mainly to briefly describe its properties such as control principle, control parameters, the possibly required sensors, block diagram and control equations.

2.3.1 Computed-Torque Control

The computed-Torque control is based on the principle of canceling the effects of gravity, friction, the manipulator inertia tensor, as well as Coriolis and centrifugal forces using the feedback signal.

This control algorithm, also known as a model-based control, was designed on the assumption that the robot is moving in open space as well as having a well

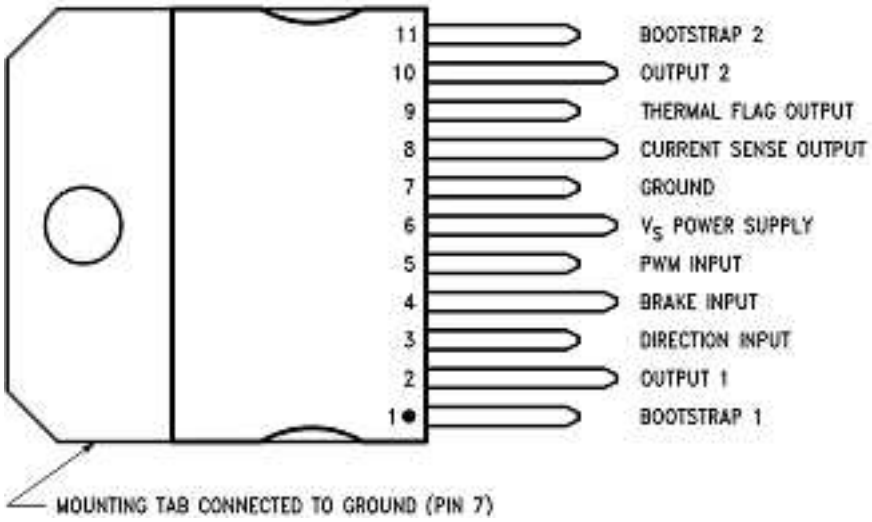

Figure 2.4: LMD18200 -with current sensor output in pin8, (Semiconductors, 2005)

known dynamic model. The dynamic model is composed of nonlinear functions of the state variables (joint positions and velocities) which characterizes the behavior of the robot manipulators. These state variables are usually measured from the following sensors: encoder, potentiometer and accelerometer.

The robot manipulator model has generally a form as in Equation 2.4 (Nguyen and Peters, 2008), and (Lewis, 2004). The robot arm dynamics are summarized in the equation below.

$$M(q)\ddot{q} + C(q,\dot{q}) + F_v\dot{q} + F_d(\dot{q}) + G(q) + \tau_d = \tau \tag{2.2}$$

$$N(q,\dot{q}) = C(q,\dot{q}) + F_v\dot{q} + F_d(\dot{q}) + G(q) \tag{2.3}$$

$$M(q)\ddot{q} + N(q,\dot{q}) + \tau_d = \tau \tag{2.4}$$

where q is the joint variable, $\dot{q}$ is the joint velocity, $\ddot{q}$ acceleration, $M(q)$ the inertia matrix of the robot, $C(q,\dot{q})$ Coriolis and centripetal forces, $G(q)$ gravity forces, τ_d the disturbance torque, τ the input torque, $F_v(q)$ the coefficient matrix of viscous friction and $F_v(q)$ dynamic friction term, and $N(q,\dot{q})$ is a vector of nonlinear feedforward compensation

A detailed analysis of this controller can be found in (Lewis, 2004). The computed-torque control algorithms are presented briefly here.. The Propor-

tional Derivative (PD) computed-torque control law is given as Equation 2.5 with the tracking error between the desired position q_d and the current position $q(t)$ defined as in Equation 2.6. However a PD controller cannot remove the steady-state error. By including the integrator in the feedforward loop removes the steady-state error. The Proportional Integrator Derivative (PID) computed-torque controller is shown as in Equation 2.7 as well as shown in Figure 2.5.

The summary of the control equation is

$$\tau = M(q)(\ddot{q} + K_v\dot{e} + K_p e) + N(q, \dot{q}) \tag{2.5}$$

$$e(t) = q_d(t) - q(t) \tag{2.6}$$

$$\tau = M(q)(\ddot{q}_d + K_v\dot{e} + K_p e + K_i\epsilon) + N(q, \dot{q}) \tag{2.7}$$

where $\dot{\epsilon}$ is equal to e, K_p is the proportional gain, K_v the derivative gain and K_i the integral gain

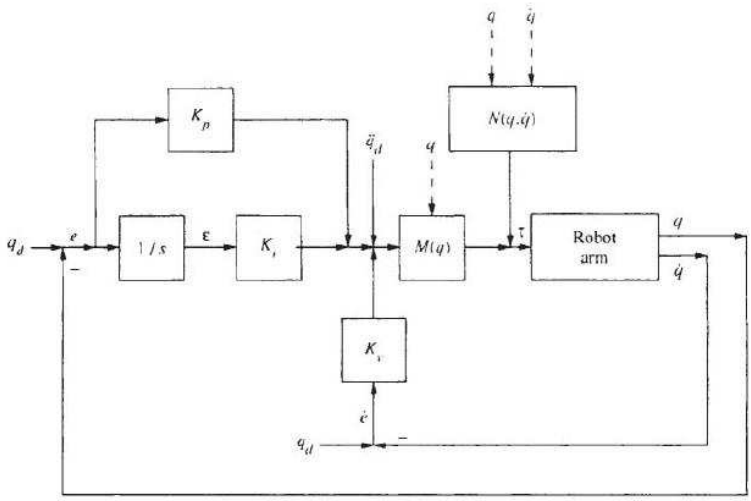

Figure 2.5: PID computed-torque controller,(Lewis, 2004)

The main disadvantages of this controller are insufficient performance due to inaccuracy in the manipulator model and time required to compute the model

(Craig, 2005). However, this controller is easy to understand and often works well in practice even though it is inconvenient to implement for PHRI, since they cannot control the manipulator when there is an unexpected force at the end effector of the manipulator.

2.3.2 Hybrid Position/Force Control

When there is force interaction at the end effector of the manipulator the computed-torque control algorithm will be unstable, therefore a different approach is needed. The use of hybrid control algorithm is one approach which was first proposed by (Hogan, 1987). It is based on decoupling the position and force control problems into subtasks.

Implementation of this algorithm requires the kinematic equation of the manipulator, the data from sensors such as from position sensors, the wrist mounted force sensors as well as the task space force derivative, nevertheless the latest signal is not usually available. If the interacted force is measured in the joint space, the manipulator dynamics equation is modified from Equation 2.4 to Equation 2.8. The detailed analysis of the manipulator dynamics equation can be found in (Lewis, 2004), (Thomas, 2005).

The modified version of the robot arm dynamics is

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F\dot{q} + G(q) + J^T(q)f \tag{2.8}$$

where $J^T(q)$ is the transpose of task space Jacobian and f is the force at the end-effector

To summarize control law, the position and the force control algorithm is given as Equation 2.9 and Equation 2.10, respectively (Lewis, 2004). Furthermore, the overall hybrid position/force control strategy is shown in Figure 2.6. The feed forward terms in the figure represent the terms in Equation 2.11.

$$a_{Ti} = \ddot{x}_{Tdi} - K_{Tvi}\dot{\tilde{x}}_{Ti} + K_{Tpi}\tilde{x}_{Ti} \tag{2.9}$$

$$a_{Nj} = \frac{1}{K_{ej}}\left\{\ddot{f}_{Ndj} + K_{Nvj}\dot{\tilde{f}}_{Nj} + K_{Npj}\tilde{f}_{Nj}\right\} \tag{2.10}$$

where

a_{Ti} is i^{th} linear tangent space position controller

x_{Tdi} is i^{th} desired position trajectory tangent to the environment surface

K_{Tvi} and K_{Tpi} are being the ith positive control gains

$\tilde{x}_{Ti}$ tangent space position tracking error

a_{Nj} is the j^{th} linear normal space force controller

K_{ej} is the j^{th} component of environmental stiffness

$\ddot{f}_{Ndj}$ is double derivative of f_{Ndj} which is the j^{th} component of the desired force exerted normal to the environment

K_{Nvj} and K_{Npj} are the jth positive control gains

f_{Nj} is j^{th} component of the force exerted normal to the environment

$\tilde{f}_{Nj}$ is equal to the difference between f_{Ndj} and f_{Nj}

f_{Nj} is j^{th} component of the force exerted normal to the environment

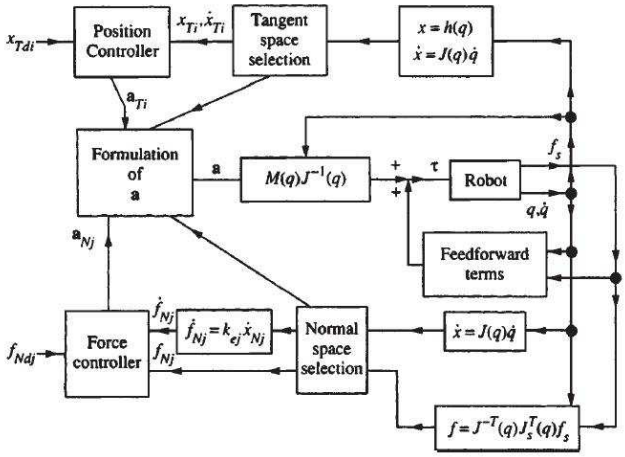

Figure 2.6: Hybrid position/force controller, (Lewis, 2004)

In Figure 2.6 terms such as $Js(q)$ is an $n \times n$ Jacobian sensor matrix, and f_s is an $n \times 1$ vector of sensor forces. The equation below shows the feed forward term in the schematic of hybrid Position/Force controller.

$$-M(q)J^{-1}(q)\dot{J}\dot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + J^T(q)f \tag{2.11}$$

Hybrid Position/Force Control is mainly used for the tracking position and force trajectories simultaneously. Most of the PHRI projects reviewed rarely use this algorithm. However (Xiao et al., 2000) demonstrate that the algorithm can be used in unknown environment by estimating the trajectory of the robot using a vision sensor.

2.3.3 Stiffness Control

(Salisbury, 1980) proposed a stiffness controller concept based on the linear spring relationship concept. That is, the robot manipulator can be visualized as a spring exerting a force on the environment. This algorithm can be easily used for force control application (Lewis, 2004). This source associated the term stiffness control with PD control, that is stiffness of the manipulator can be tuned by adjusting K_p as well as damping gain.

$$f = K_p(x_d - x_e) \tag{2.12}$$

where f is the force exerted on the environment, K_p the desired stiffness of the manipulator, x_d the desired position and x_e initial position of the manipulator.

The implementation of this algorithm needs to sense the force. Another source (Salisbury, 1980) show that, this sensor can be place at the actuator, at the wrist, at the fingers or in the environment which the manipulator is in contact with.

(Lewis, 2004) summarized this controller as in Equation 2.13. This controller is a PD-type controller where K_v and K_p are NxN diagonal where N stands for number of joints, constant, positive-definite matrices and the task space of end-effector tracking error $\tilde{x}$ is defined as in equation 2.14.

$$\tau = J^T(q)(-K_v\dot{x} + K_p\tilde{x}) + G(q) + F(\dot{q}) \tag{2.13}$$

$$\tilde{x} = x_d - x \tag{2.14}$$

This controller can be used in PHRI as illustrated in the work of (Kosuge and Kazamura, 1997). The basic limitation of this algorithm is that the desired end effector manipulator position and the desired force exerted on the environment must be constant (Lewis, 2004).

2.3.4 Impedance Control Algorithm

The Impedance control algorithm, analogous to Ohm's law, is based on the concept that the controller should be used to regulate the dynamic behavior between the robot manipulator motion and the force exerted on the environment. This algorithm considers the motion and force control problem together unlike hybrid position/force control (Hogan, 1987).

To implement this algorithm, a environmental impedance model is required and also selection of appropriate manipulator impedance as well as measurement of force, position and speed of the end-effector. The manipulator impedance is selected to have zero steady-state error for a step input command of force or velocity based on the duality principle (Thomas, 2005).

This algorithm does not need to switch from position control mode to force control mode unlike hybrid position/force control. In addition, it permits to define priori the way the manipulator interacts with the unknown environment (Morel et al., 1998). However, hybrid control has a good performance to control precisely the position and force trajectories of the end-effector (Yoshikawa, 2000).

A summary of the control law is presented here. More detailed analysis can be found in (Thomas, 2005), (Lewis, 2004). The torque control law equation is the same as Equation 2.8. The position and force control law is shown in Equations 2.15 and 2.16 respectively. The overall control strategy are shown in Figure 2.7.

The summary of the impedance control equation is

$$a_{pi} = L^{-1}\left\{s(\dot{x}_{pdi}(s) - Z_{pmi}^{-1}(s)f_{pi}(s))\right\} \tag{2.15}$$

$$a_{fj} = L^{-1}\left\{sZ_{fmj}^{-1}(s)(f_{fdj}(s) - f_{fj}(s))\right\} \tag{2.16}$$

where subscript p denotes position control and subscript f denotes force control, a represents the impedance position and force control strategies, Z_{pmi} and Z_{fmi} is the i_{th} position and force controlled manipulator impedance respectively. L^{-1} is used to represent the inverse Laplace transform operation, f the environmental forces.

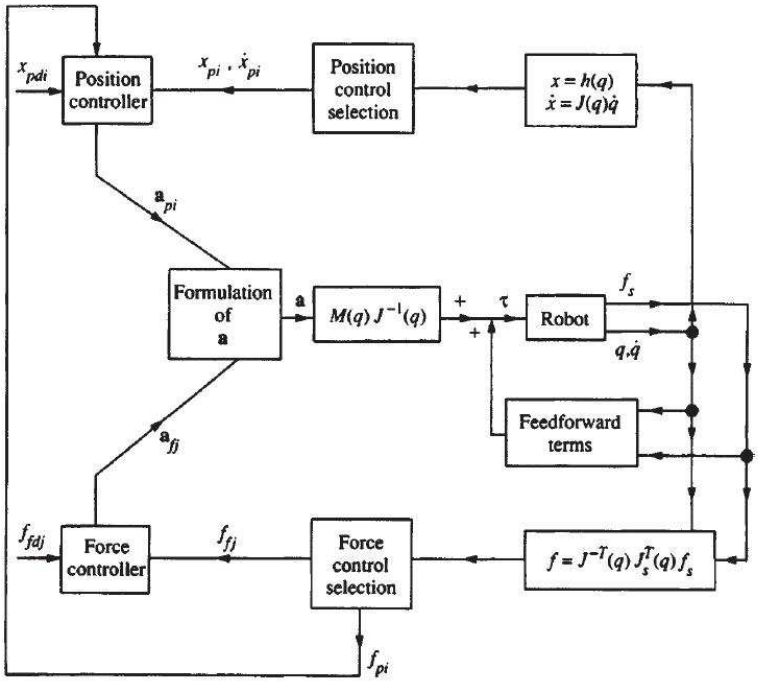

Figure 2.7: Impedance controller, (Lewis, 2004)

One of application areas of this algorithm is for PHRI because a compliant behavior of the manipulator leads to safe physical interaction as demonstrated in the work of (Albu-Schaffer et al., 2008), (Kosuge and Kazamura, 1997), (Tsumugiwa et al., 2002). In addition, (Morel et al., 1998), (Morel et al., 1998), (Siciliano and Villani, 2007) also used this algorithm in cooperation with vision sensing for cooperation tasks between humans and robots.

2.4 Summary and Conclusions

This literature review is conducted with the aim of assessing astronaut-robot physical interaction in object manipulation. The state-of-the-art of PHRI were covered, and very little was found in the literature of astronaut-robot cooperation for the purpose of manipulating object. There are, however, many works related to human-robot physical interaction which are applicable for astronaut-cooperation. In addition, the commonly used sensors for force control and manipulator force control algorithms were reviewed. Appendix A contains the summary of the algorithms and sensors covered in this literature review.

The use of good sensors is very vital to improve the efficiency of the force algorithms as well as most importantly for the safety of the astronaut to work with in cooperation with the robot. For instance, the use of vision sensors together with force/torque sensors will make the PHRI safer by avoiding unexpected human interaction and increase the efficiency of the force control algorithms. However, vision sensors and force/torque sensors measure two different states (contact force and vision) and insufficient work has been done to fuse these data to yield one state.

In this review six axis force/torque sensors were found to be more accurate and reliable senor compared to the other options available. However, more testing on these sensors need to be undertaken to use it for space missions. Likewise, the impedance, admittance and stiffness control algorithms are potential algorithms to control robots for physical interaction tasks between astronaut-robot.

Chapter 3

Design of WorkPartner Manipulator Compliant Control

"If something is worth doing once, it's worth building a tool to do it." by Tom Van Vleck

WorkPartner is a mobile service robot, looks like a centaur so it may be called centaur, designed and implemented in the department of automation and systems technology, Helsinki University of Technology (TKK). It has a human-like two hand manipulator which enables it to work with everyday life tasks in an outdoor environment. Figure 3.1 shows the complete structure of *WorkPartner*.

WorkPartner interacts using different interfaces such as speech, gesture, and visual interfaces. The applications of *WorkPartner* can be extended for space missions due to a suitable hybrid locomotion system. The locomotion system allows motion with legs and/or wheels at the same time depending on the environment condition (of TKK, 2005).

Figure 3.1: Robot concept illustration and the real robot (of TKK, 2005)

3.1 Hardware Description

WorkPartner has six degrees of freedom, i.e it can reach any positions and orientation in three dimensional space, which results in advanced flexibility in comparison to standard industrial robots. It has a motor position sensor and a sensor for joint position. Table 3.1 summarizes the hardware specification of *WorkPartner*. Figure 3.2 shows that a single side manipulator joint angle range constraint of the *WorkPartner* robot comprises of five revolute joints, of which two are used for wrist motion (use for inclination and rotation), one is for elbow and two are used for a shoulder inclination and yaw rotation motion.

The joints are controlled by PID controller which has limited access for modification and integration with the main control system. Due to these limited access of the controller and insufficient performance to add additional control algorithms, it is changed into a new commercial controller called Elmo driver (Elmoc, 2009). This controller is programmed using windows operating system computer connected through a Controller Area Network (CAN) bus or RS-232 communication.

Specification	*Description*
Max. payload	10kg
Number of axis	5 Revolute Joints
Maximum Reach	1 m
Motors	Graphite Brushes, Solenoid
Gears Ratio	957.6:1 to 2272.4:1
Sensors (each Joint)	Potentiometer and Encoder
Brakes	Solenoid Motors Controlled by FET
Power Supply	48V/4.15A and 12V for Brakes
Control	Position control (PID)
New Control Drive	Elmo Drive
Communication	CAN and RS-232

Table 3.1: Hardware specification of *WorkPartner* manipulator

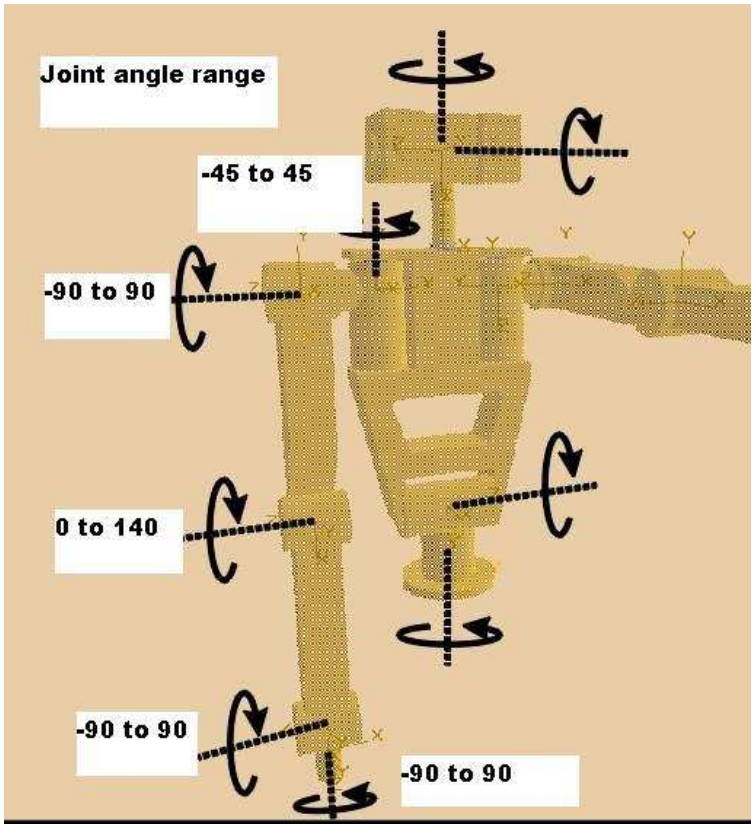

Figure 3.2: Ranges of *WorkPartner* Joint angle (of TKK, 2005)

3.2 Elmo Controller

Two Elmo Whistle commercial controllers is bought and tested for position control of *WorkPartner* manipulator . These controllers support both RS-232 and CAN open communication. In addition, these commercial controllers have features such as current control, velocity control and position control modes (Elmoc, 2009). Figure 3.3 shows the controller architecture of the Elmo controller. The controller is modified as shown in Figure 3.4 that enables to use old controller connectors with sensors and motors of the manipulator joints.

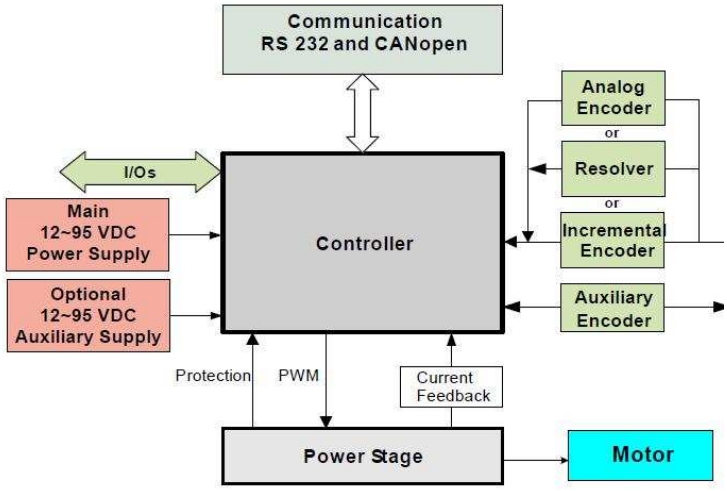

Figure 3.3: Elmo Whistle controller system architecture (Elmoc, 2009)

3.3 *WorkPartner* Manipulator Forward Kinematics

The kinematics analysis of the *WorkPartner* enables computing the position and orientation of the manipulator's end-effector relative to the base of the manipulator as a function of the joint variables. Denavit-Hartenberg convention is used to describe the kinematic mechanism of *WorkPartner* using the link parameters. The parameters of Denavit-Hartenberg are revolute joint θ_i, link

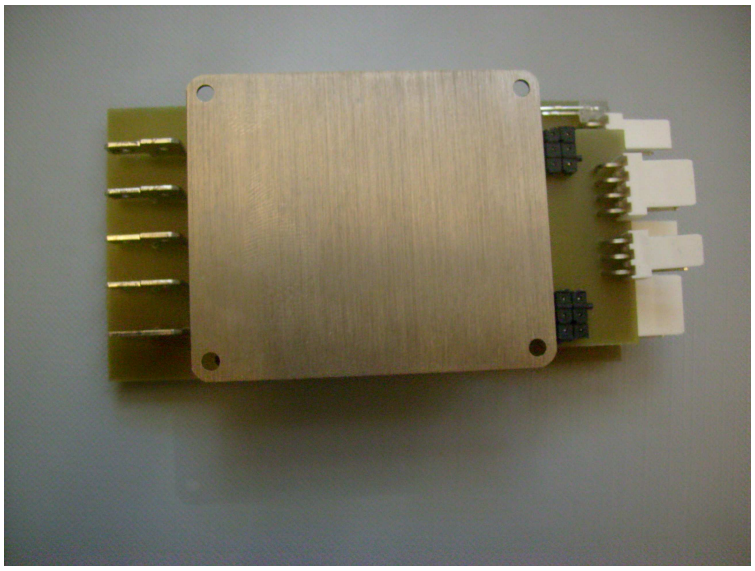

Figure 3.4: Elmo controller after circuit modification

length a_i, link twist α_i, and link offset d_i. In case of prismatic joint the link offset d_i is the joint variable and other three are fixed (Craig, 2005). Table 3.2 shows link parameters for the *WorkPartner* which is made based on the link-frame attachment shown in Figure 3.5.

Where the parameters on the table 3.2 are a_{i-1}: the distance from $\hat{Z_{i-1}}$ to $\hat{Z}_i$ measured along $\hat{X_{i-1}}$
α_{i-1}: the angel from $\hat{Z_{i-1}}$ to $\hat{Z}_i$ measured about $\hat{X_{i-1}}$
d_i: the distance from $\hat{X_{i-1}}$ to $\hat{X}_i$ measured along $\hat{Z}_i$
θ_i: the angel from $\hat{X_{i-1}}$ to $\hat{X}_i$ measured about $\hat{Z}_i$

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_i
2	-90	0	0	θ_i
3	0	0.3	0	θ_i
4	0	0.3	0	θ_i
5	-90	0	0	θ_i

Table 3.2: Link parameters for the *WorkPartner* manipulator

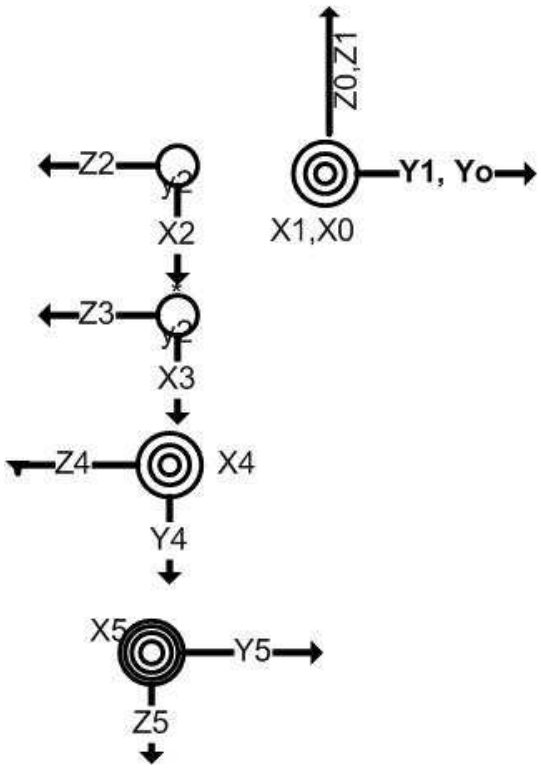

Figure 3.5: Link frame attachment of *WorkPartner* manipulator

To compute the kinematic equation the transformation matrices method of Denavit-Hartenberg is used. For each adjacent joint a transformation matrix is calculated by using the general form of transformation matrices given in Equation 3.1 (Craig, 2005).

$$
{}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i_1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i_1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1}
$$

where c is cos() and s is sin(). The kinematic equation is computed from the values of the link parameters, the individual link-transformation which can be computed using Equation 3.1. Equation 3.2 is used to find the final single transformation that relate frame N to from 0.

$$
{}^{0}_{N}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T \cdots {}^{N-1}_{N}T \tag{3.2}
$$

3.4 *WorkPartner* Manipulator Inverse Kinematics

The inverse kinematics, which means the calculation of the joint parameters to reach a given point in space with the end effector (pose), for the *WorkPartner* is described here. Multiple techniques exist to do these calculations such as geometric, algebraic, damped least squares and dual quaternion methods.

This section presents two type of methods to solve the inverse kinematic problem. These are geometric and damped least square methods. The first method, geometric method, is implemented on the two joint arm SimPartner simulator to demonstrate the chosen control algorithm to perform the cooperation task. The other, damped least square, is an algorithm for robot arms with more than three degree of freedom.

3.4.1 Geometric Method

In this thesis SimPartner, the *WorkPartner* simulator, is modified to suit for testing the control algorithm, such as force sensors and position sensors are added on shoulder, elbow and wrist joints of one arm of the SimPartner (Heiskanen et al., 2008) . Figure 3.6 shows the screen shot of the simulator developed for *WorkPartner*. The client can send commands such as request sensor information and set the joint actuators parameters on the corresponding sensors and actuators. The server replies the requested information or replies status message to the corresponding command.

The SimPartner arm frame of reference is shown in Figure 3.7. The joints are referenced to a global frame of reference. These frames are transformed from inertial frame of reference to the shoulder frame using Equations 3.3 and 3.4 by computing the inverse kinematics.

$$ {}^{3}_{1}T = {}^{0}_{1}T \, {}^{2}_{0}T \tag{3.3} $$

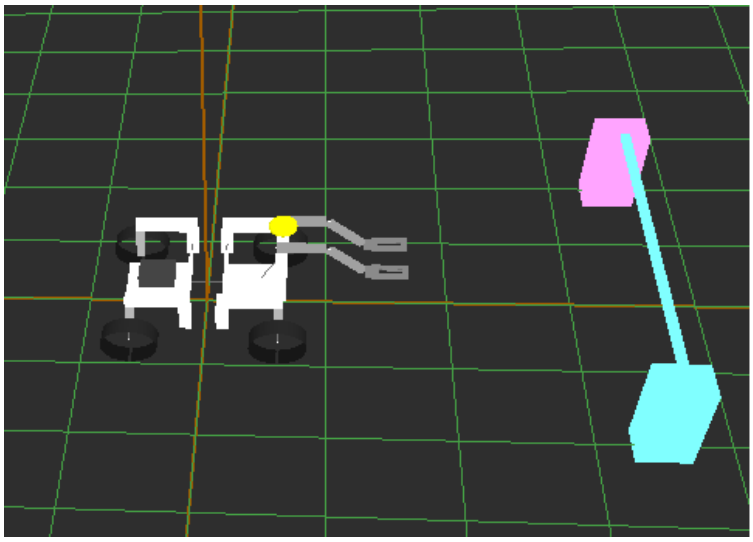

Figure 3.6: *WorkPartner* simulator screen shot used for this thesis before modification

$$^{2}_{1}T = ^{0}_{1}T . ^{3}_{0}T \tag{3.4}$$

The inverse of the transform is computed using the full advantage of the structure of homogeneous transform from Equation 3.4.1. This equation gives description of A relative to B - that is, $^{B}_{A}T$. (Craig, 2005)

$$^{B}_{A}T = \begin{bmatrix} & ^{A}_{B}R & & -^{B}_{A}R^{T}.P_{B^{A}_{ORG}} \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

After the frames are transformed from the inertial reference frame to the shoulder reference frame, the frames are shown in Figure 3.7. The end effector position is computed with reference to the shoulder frame. This allows computing the angular position of the shoulder joint, θ_1 ,and the angular position of the elbow joint, θ_2, from the end-effector linear position x_e and y_e using geometric method.

$$x_e^2 + y_e^2 = L_1^2 + L_2^2 - 2L_1L_2cos(\pi - \theta_2) \tag{3.5}$$

where L_1 and L_2 are the length of upper arm and lower arm respectively.

$$-2L_1L_2\cos(\pi-\theta_2) = x_e^2 + y_e^2 - ({L_1}^2 + {L_2}^2) \tag{3.6}$$

$$-2L_1L_2\cos(\theta_2) = x_e^2 + y_e^2 - ({L_1}^2 + {L_2}^2) \tag{3.7}$$

$$\cos(\theta_2) = \frac{x_e^2 + y_e^2 - ({L_1}^2 + {L_2}^2)}{2L_1L_2} \tag{3.8}$$

$$\tan(\psi) = \frac{L_2 s_2}{L_1 + L_2 c_2} \tag{3.9}$$

where s_2 is $\sin\theta_2$, c_2 is $\cos\theta_2$,s_1 is $\sin\theta_1$, c_1 is $\cos\theta_1$.

$$\theta_1 = \arctan(\frac{y_e}{x_e}) - \psi \tag{3.10}$$

Simplifying Equations 3.8, 3.9 and 3.11 gives θ_1 and θ_2.

$$\theta_1 = \arctan(\frac{y_e}{x_2}) - \arctan(\frac{L_2 s_2}{L_1 + L_2 c_2}) \tag{3.11}$$

$$\theta_2 = \arctan([s_2, s_1]) \tag{3.12}$$

where

$$c_2 = \left(\frac{{x_e}^2 + {y_e}^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right) \tag{3.13}$$

and

$$s_2 = \pm\sqrt{1 - c_2} \tag{3.14}$$

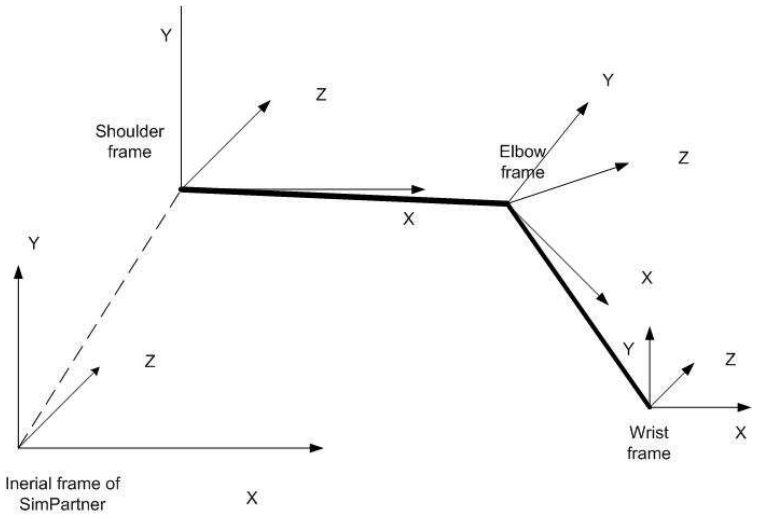

Figure 3.7: Frame of reference of SimPartner simulator

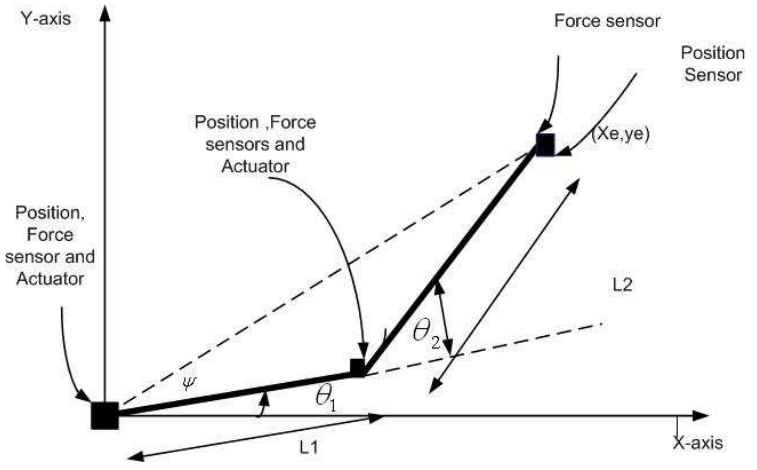

Figure 3.8: SimPartner arm configration

3.4.2 Damped Least Squares Method

Most of the inverse kinematics methods has inefficient performance around the vicinity of singularities. Singularity occurs - when the manipulator is fully stretched out or folded back on itself, as well as when two or more joint axes are lining up (Craig, 2005). This problem should be avoided because the high speed of some parts of the robot while passing these singularities is dangerous to the human partner while human and robot are in a cooperation task.

Damped Least Squares (DLS) works by finding the value of $\Delta\vec{\theta}$ that minimizes the quantity $||J\Delta\vec{\theta} - \Delta\vec{X}||^2 + \lambda^2||\Delta\vec{\theta}||^2$, where $\lambda > 0$ is a non-zero damping constant which depends on the details of the multi body and the target position.The lambda has to be chosen carefully to make DLS numerically stable and $\Delta\vec{X}$ is the linear position error. This is calculated in Equation 3.15.

To summarize the DLS method avoids problems around singularities by introducing a damping constant. The DLS matrix which is an *nxn* matrix where n is the number of degrees of freedom, can be expressed in Equation 3.16. This damping constant should be large enough so that the solutions for $\Delta\vec{\theta}$ are well-behaved near singularities but not so large that the convergence rate is too slow (Buss and Kim., 2004).

The DLS matrix can be expressed in equations below.

$$\Delta\vec{\theta} = J^T(JJ^T + \lambda^2 I)^{-1}\Delta\vec{X} \tag{3.15}$$

$$J^T(JJ^T + \lambda^2 I)^{-1} = \sum_{i=1}^{r} \frac{\sigma_i}{\sigma_i^2 + \lambda^2} v_i u_i^T \tag{3.16}$$

where J is the Jacobian matrix, r is the rank of J, the columns of the matrices are defined as u_i of U and v_i of V respectively, σ_i, U and V are taken from the Single Value Decomposition (SVD) of J, and λ is damping constant.

A MATLAB simulation of DLS is implemented using *WorkPartner* direct kinematic equation. The steps followed to solve inverse kinematic using DLS are:

1. Compute the SVD from the Jacobian calculated from the direct kinematics.
2. Compute the difference between goal position and current position of the robot end effector which is $\Delta\vec{X}$.
3. Compute the change in angle, $\Delta\vec{\theta}$, of each joint using DLS algorithm and update the value of each joint accordingly.
4. Iterate through these steps until the required target is reached.

However, this algorithm needs to read the current position of the robot as fast as possible and position command should be sent quickly enough which makes the simulator barely stable, if this algorithm is implemented on SimPartner. This instability could be due to the client needing to send a request command for each sensor data and get a reply from the server. This problem could be avoided by implementing the algorithm to server side.

3.5 *WorkPartner* Manipulator Control System Design

The realization of compliant behavior is relevant whenever the robot comes into contact with its environment and especially if there is insufficient knowledge about the environment. The *WorkPartner* manipulator uses a PID controller to control the trajectory. However this controller has limited access for upgrading and further development works such as for compliant control.

In this thesis compliant control strategy is required for a *WorkPartner* robot manipulator to be able to interact with human safely. To achieve this goal the impedance control algorithm is suggested based on the literature review. The aim of an impedance control methods are to establish a mass-damper-spring relationship between the Cartesian position $\Delta\chi$ and the Cartesian force or torque F:

$$F = M\Delta\ddot{\chi} + D_k\Delta\dot{\chi} + K_k\Delta\chi \tag{3.17}$$

where M, D_k and K_k are all positive definite matrics representing virtual inertia, the damping factor and stiffness of the system (Albu-Schaffer and Hirzinger, 2002). Figure 3.9 shows the one dimensional model of mass-damper-spring model.

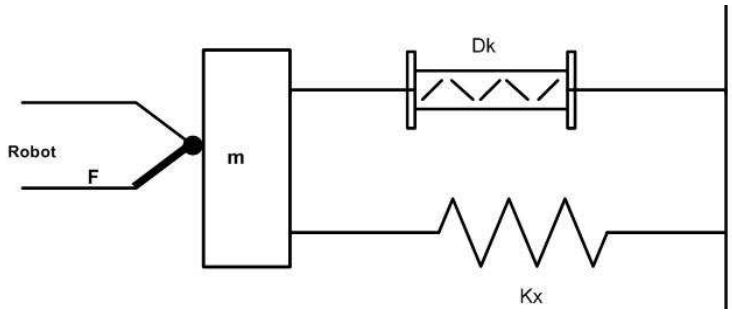

Figure 3.9: Model of environmental impedances

In addition to Chapter 2 review, brief introduction of the suggested admittance control and impedance control algorithm are presented in the following section.

3.5.1 Impedance Control

Impedance control obtains a dynamic relationship between position and force, rather than controlling either of these variables alone (Hogan, 1987). More detailed information about the impedance control algorithm can be found from Chapter2. The impedance control uses directly the equation 3.17, where the actual Cartesian position χ is computed from the joint angular position q using direct kinematics,$K(q)$

$$\chi = K(q) \tag{3.18}$$

Equation 3.19 allows the conversion of the desired joint torques τ_d to the Cartesian force-torque vector by using the transposed Jacobian $J^T(q)$. Then the motor τ_R transforms the motor torque commands τ_m from the desired joint torques as indicated in Equation 3.20.

$$\tau_d = J^T(q)F \tag{3.19}$$

$$\tau_m = \tau_R(\tau_d) \tag{3.20}$$

Using the transposed Jacobian $J^T(q)$, the Cartesian force is transformed into the desired joint torques. Figure 3.10 shows the simplified schematic of impedance control. This control method requires a joint torque. In addition due to disturbing influence of friction, direct current command has limited application as a torque control or impedance control (Albu-Schaffer and Hirzinger, 2002).

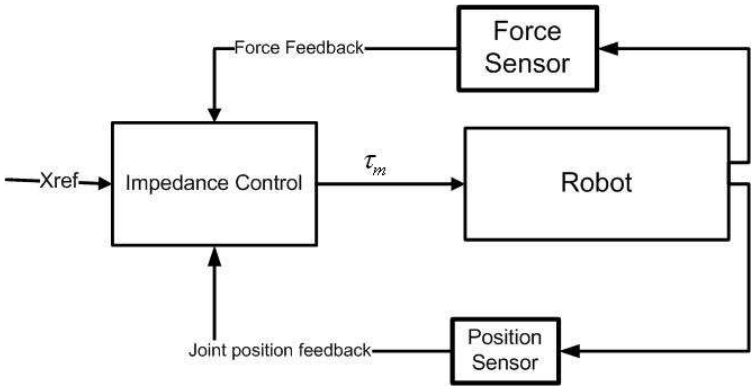

Figure 3.10: Impedance control block diagram

3.5.2 Admittance Control

Admittance control is the inverse of impedance control. Figure 3.11 shows the schematic of admittance control where F_d is desired force and x_{ref} is a reference position. In the case of admittance control, the force is measured at the end-effector using six-axis force/torque sensor. This force vector, $\Delta f(s)$, is utilized to generate a desired Cartesian position χ_d (Albu-Schaffer and Hirzinger, 2002).

$$\chi_d(s) = \chi_0(s) - \frac{\Delta f(s)}{K + Ds} \tag{3.21}$$

where $\chi_0(s)$ is initial position, K is stiffness constant and D is damping constant. This Cartesian position is converted to the desired joint positions using the inverse kinematics.

$$q_d = K^{-1}(x_d) \tag{3.22}$$

Then the joint position controller P_R generates the motor torques.

$$\tau_m = P_R(q_d) \tag{3.23}$$

This method is widely used, since a position control interface is available on most robotic manipulator systems (Albu-Schaffer and Hirzinger, 2002).

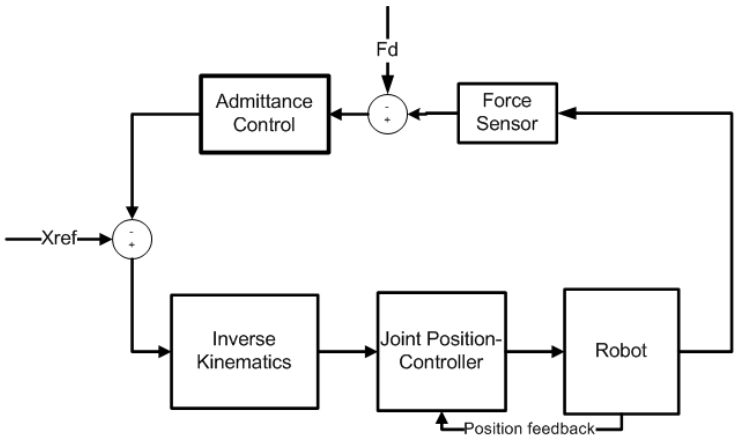

Figure 3.11: Admittance control schematic

3.5.3 Control Approach

In this thesis position-based impedance control and admittance control, is implemented which enables the use of the commercial controller called the Elmo controller. Elmo controller has position control interface in addition to other features. The Elmo position controller is similar to the other position controller and has an advantage of being able to compensate for friction in the joints (Albu-Schaffer and Hirzinger, 2002). In addition this control approach works both for static and moving *WorkPartner* since the algorithm uses the environment model not the *WorkPartner* dynamic model.

The test setup of the proposed control system is shown in Figure 3.12. Figure 3.13 shows the control algorithm scheme to be implemented on the *WorkPartner*.

The position-based impedance control algorithm allows the robot to interact both in constrained and unconstrained areas. The control concept which is tested on the SimPartner simulator has followed the steps:

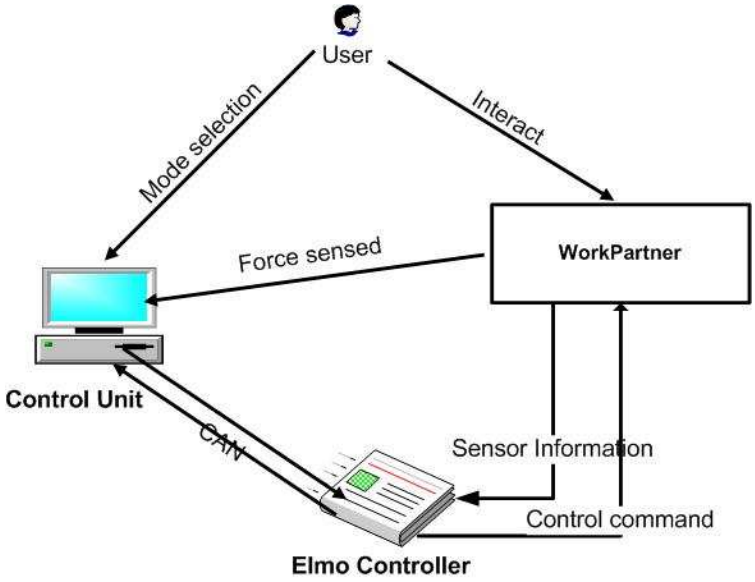

Figure 3.12: Prototype of control system

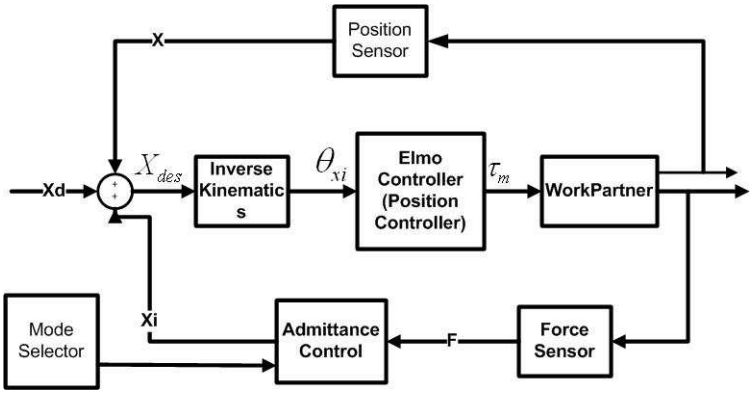

Figure 3.13: Proposed control algorithm based on admittance control

- The external force, F, is measured using force sensor (directly measured by means of the sensor mounted at the manipulator endpoint and reported to the control unit)

- The impedance (admittance) control algorithm determines the next end-effector linear positions as shown in Equation 3.24 from the measured force vector, based on the mode of operation. The mode of operation determines possible range of values of stiffness constant. For example follow movement expects low value of stiffness gain so that the robot will follow the direction of force applied on it.

$$\chi_{des}(s) = \chi(s) + \chi_d(s) - \frac{\Delta f(s)}{K} \tag{3.24}$$

  where $\chi(s)$ is relative position of end-effector, $\chi_i(s)$ is absolute position from the control algorithm and K is stiffness constant.

- Using the inverse kinematics, linear position from the previous step, the desired position and the position output from the admittance control, the algorithm transforms into the angular positions of each joint as in Equation 3.25.

$$q_d = K^{-1}(\chi_{des}) \tag{3.25}$$

- Position controller uses the angular positions for each joint from the previous step to generate a corresponding torque command for each of the robot joints.

Chapter 4

Compliance Control Algorithm Implementation

"Everything should be made as simple as possible, but not simpler." Albert Einstein

This chapter examines some possible cases of astronaut-robot cooperation to perform manipulation tasks. In addition, the chapter describes the implementation of the chosen algorithm for safe interaction. Modified version of SimPartner, *WorkPartner* simulator, is used to test the control algorithm.

4.1 *WorkPartner* Simulator Modification

SimPartner is an object-oriented dynamic robot simulator which has been developed based on Open Source projects such as Open Dynamic Engines. This simulator is modified to suit the requirements for this thesis demonstration and to increase the performance. The existing simulator supports adding the required new components such as force sensors, actuators, position sensors. However the performance is affected by each additional component. To meet those requirements, SimPartner is modified and the required components are

added mainly for the left arm of *WorkPartner* such as:

- force sensors on the shoulder, elbow and wrist joints.
- position sensors on the shoulder, elbow and wrist joints.
- the robot model is simplified to increase the processing speed.
- an object is added to the environment that interacts with the robot model and is able to apply force.
- a position actuator is added for the object added to the environment.
- the inertial reference frame is transformed to the shoulder frame.

Figure 4.1 shows simplified SimPartner interacting with an object that has position actuator.

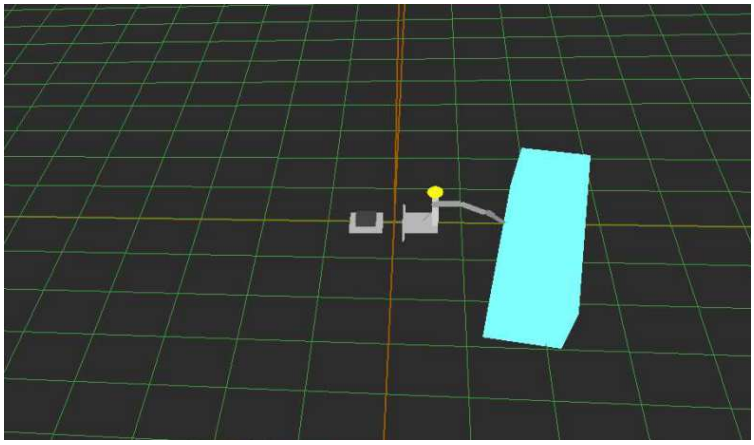

Figure 4.1: SimPartner interacting with object

4.1.1 Software Design

Brief description is presented here for the main classes modified from SimPartner simulator and the new classes.

Admittance class
This class implements the admittance control algorithm which takes force inputs and then generate the corresponding linear position.
InverseKinematics class
This class implements the geometric methods of inverse kinematics of SimPartner manipulator.
ClientInterface
This is the main class that implements the client interface to receive and send sensor information, command data as well as status message from each component of the simulator. In addition this main class implements follow movement, hold position as well as adapt movement demo cases. Asynchronous communication is used to communicate between the client interface and the server interface.
SimPartner
This is the main class of server side of the simulator only few modification made to include the force sensors and position sensors.
ForceSensor class
This sensor is implemented based on Open Dynamics Engine (ODE) model to measure the joint force/torque which returns the force in each axis. The sensor return the force/torque only at request.

4.2 Physical Manipulator Interaction Demos

In order to tests the developed control algorithm performance, four different PHRI modes were selected for examination. Each of the modes,shown in 4.3 are presented with there respective results.

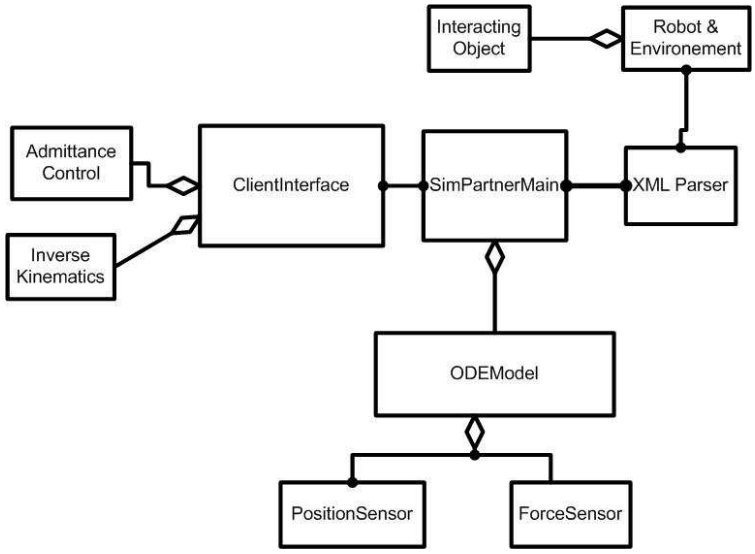

Figure 4.2: Simplified class diagram of simpartner

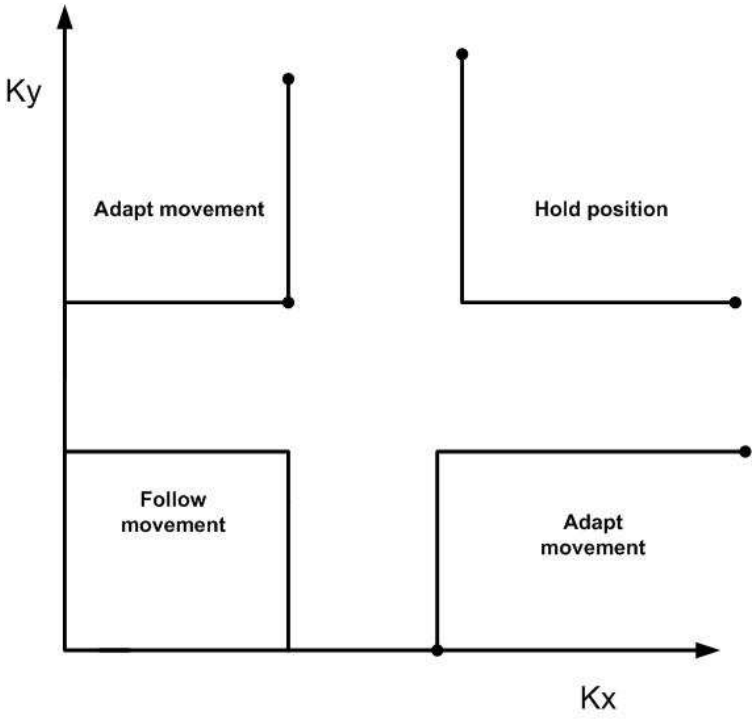

Figure 4.3: Mode of operation of the new control algorithm

4.2.1 Follow Movements

The follow movement demonstration case describes the possibilities of the astronaut to lead the robot arm to a target location using little force/torque. This demonstration is implemented on SimPartner using admittance control without a damping constant which is the stiffness control algorithm because the speed of the end-effector is unavailable. The application is developed in such away that the user is able to choose this mode of operation by keyboard. However, in a practical case this could be done using different interfaces such as speech. In a follow movement case, the user can choose the stiffness constant values K_X and K_y approximately in the range shown in Figure 4.3. These linear position values will be converted to corresponding joint angles using inverse kinematics. Figures 4.4 and 4.5 show the position change error using different stiffness values which then generate joint commands to follow the applied force. The results from the graphs are not ideal due to variation of interaction forces on both x-axis and y-axis. This creates deviation in position error from the ideal case.

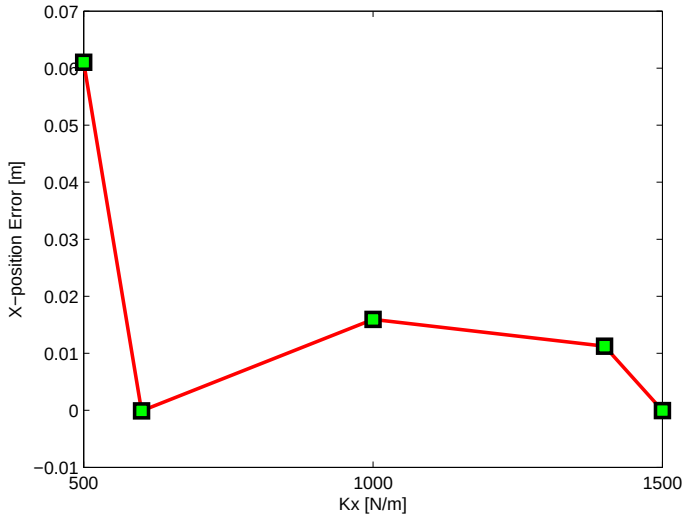

Figure 4.4: X-position change, from initial position to final position due to external force, vs stiffness Kx

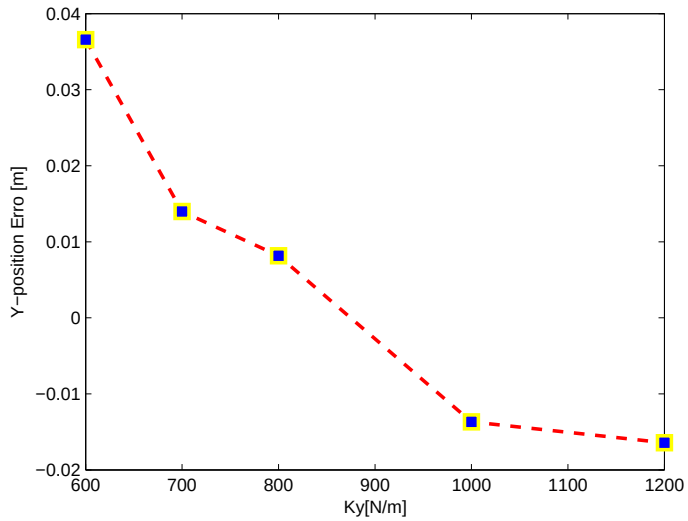

Figure 4.5: Y-position change, from initial position to final position due to external force, vs stiffness Ky

4.2.2 Hold Position

The hold position demonstration case also describes one mode of operation that enables the human to keep the manipulator in the desired position. This is demonstrated by controlling the position of the manipulator while keeping manipulator stiff when there is external force applied at the end-effector. On the *WorkPartner*, manipulator end-effector can be held in position by keeping the Elmo controller in position controller mode and sending the required positions for each joint.

The algorithm for this demonstration case is implemented on SimPartner simulator. Similarly, the user is allowed to choose the mode of operation and then give stiffness values in the approximated range as shown in Figure 4.3 which are large values of K_X and K_y. These high values of stiffness are converted to very small values of linear position which will hardly change the current position. The higher the value of the stiffness, the more accurately the end-effector holds the previous position. Figure 4.6 shows the position error using different values of stiffness K_x and K_y which have a small error when the stiffness value is higher.

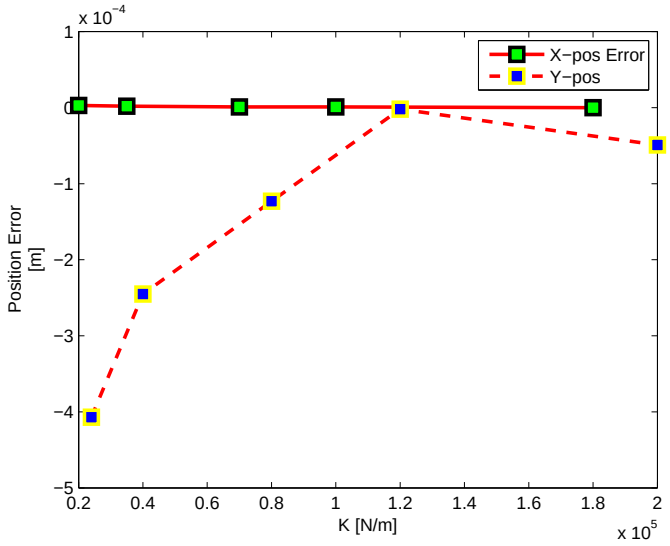

Figure 4.6: X and Y -Position change, from initial position to final position due to external force, with high values of stiffness Kx, Ky

4.2.3 Adapt Movements

The adapt movement demonstration case illustrates the possibilities of the human to be able to lead the robot arm to a target location in one direction and keep the others constant using little force/torque. This demonstration is also implemented on SimPartner using admittance control without damping constant which is the stiffness control algorithm. Like the follow movement case, this demonstration is developed in such away that the user allows to choose adapt movement mode of operation from keyboard. After that the user can choose appropriate stiffness constant. For example if the user wishes to adapt the movement in the x-direction by keeping y-direction constant, the application expects a high value for K_y and a small value in K_x . Figure 4.3 suggests the approximate range of values for the adapt movement case. After the stiffness constant the algorithm generates other values such as linear position and angular joint position and sends the command signal to the actuators. Figures 4.7 and 4.8 show the simulation result with different values of stiffness which shows the error decreased as the values increase. In this way, the movement can be kept minimized in one direction with respect to the other direction.

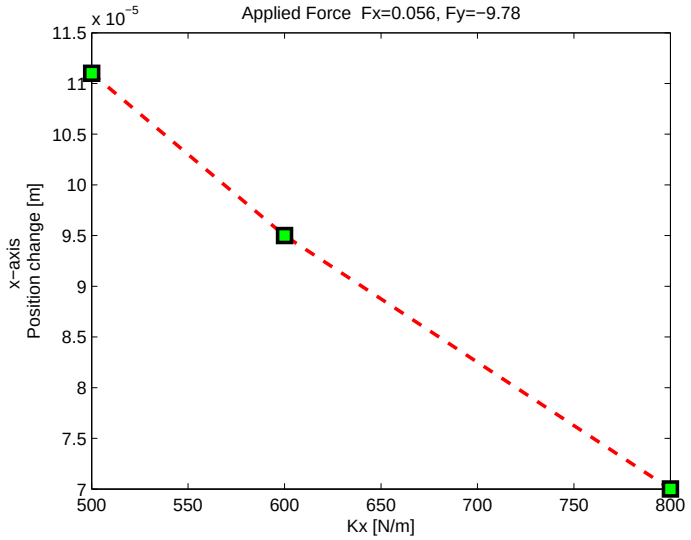

Figure 4.7: Adapt movement in X-direction

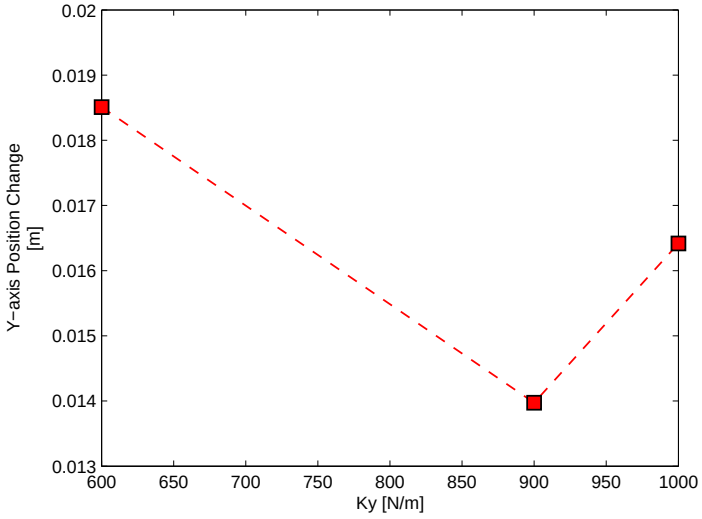

Figure 4.8: Adapt movement Y-direction

4.2.4 Push with Force

Push with force means that the robot is asked to push the object with certain force. This mode of operation is implemented after the manipulator is kept near to the object to be pushed. The interacting user sends the end-effector next position in the direction of the object to be pushed. The motion of the end-effector to the destination point should be short enough that allows straight line trajectory. Then the manipulator push the object until the end-effector force sensor senses the required force. The push with force mode was implemented on the simulator, but due to time constraints, the tests were not finally done. Basically the push with force mode is inverse case of the hold position mode, i.e. certain position errors generates a force, based on the chosen K, in push with force mode, while certain force generates a position error in hold position. For this reason the algorithm should work also for the push with force mode as it operates using basically same lines of code.

4.2.5 Discussion

The control algorithm implemented on the SimPartner losses it stability in the conditions such as at singularity and when there is applied excessive force by external object. One cause of this instability, singularity, is when the computed linear position exceeds the range of workspace of the manipulator. The application handles this exception by keeping the previous position and neglect the new command generated from the applied force.

The other cause of instability is when the excess force moves the whole robot instead of the manipulator. In real robot this could also damage the manipulator when the applied force in each joint exceed the actuator maximum force limits due to interaction from external excess force at the end effector.

4.3 *WorkPartner* Implementation

As indicated in the previous sections, stiffness control algorithm is implemented on the simulator and results plotted for each demonstration case. To implement the chosen algorithm on the *WorkPartner*, it requires to have five joint controllers and force/torque sensor at the end-effector. However the available resources are the two joint position controllers which are mounted at the shoulder and elbow of an arm of the *WorkPartner*. These controllers are tested using visual C++ programming through CAN and RS-232 communication protocol.

Based on literature review six-axis force/torque sensors is suggested. Nevertheless this sensor is expensive which is beyond the thesis budget. In spite of that the end-effector force is approximated using the controller active current. This current value is analyzed to relate the end-effector force and change in current due to impact or load at the end-effector. The detail analysis and result is presented in the next chapter.

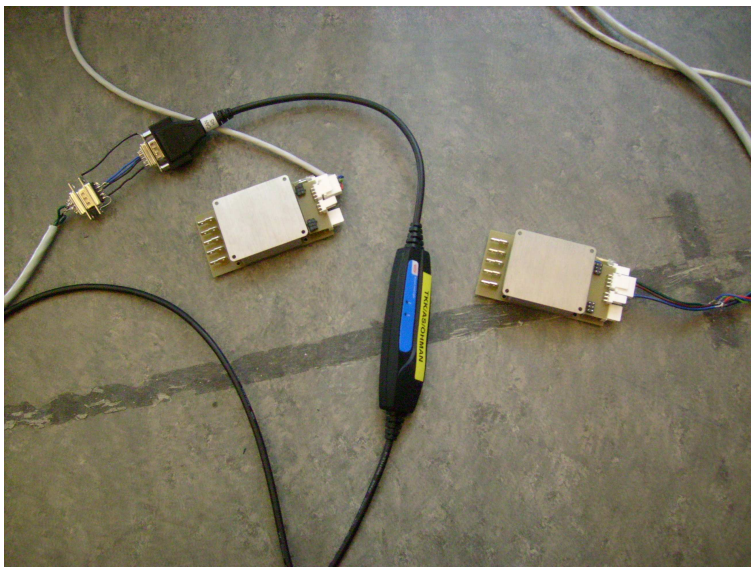

Figure 4.9: Elmo controllers and CAN device configuration

Figure 4.9 shows the Elmo controllers and Kvaser CAN driver picture before mounted. Using this driver and CAN driver the hardware implementation has done on the Elbow to demonstrate the follow movement and hold position mode of operation . The current versus force relation is done for one joint

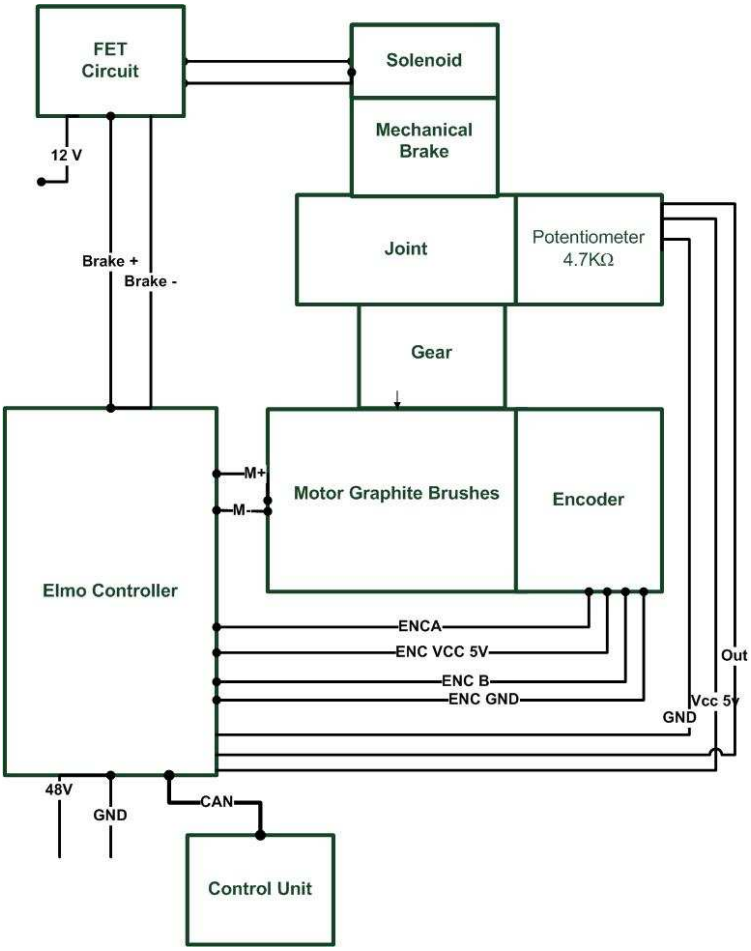

Figure 4.10: Elmo controller and *WorkPartner* joint interface

which restricts to illustrate adapt movement mode. Figure 4.10 shows the connection on the manipulator joint between the Elmo control drive, motors, sensors, brake circuit as well as *WorkPartner* manipulator joint gear.

Figure 4.11 shows effect of the stiffness change with respect to the position error when $3.11kg$ object is put at the end-effector of the manipulator. When the value of the stiffness is increased, the stiffness of the manipulator increases which changes the state from follow movement to hold position. When the stiffness is changed from 100 to 40000, the manipulator is changed from follow movement to approximately to hold position mode. This result demonstrates the two modes of operation.

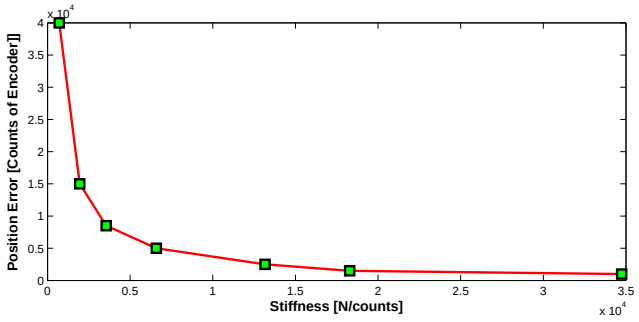

Figure 4.11: Stiffness versus position error

4.4 *WorkPartner* Tests and Results

An experiment is done to investigate the possibilities of measuring force using the active current from the Elmo controller. This controller is tested on the experiment setup shown in Figure 4.12.

4.4.1 Testing Configuration

Elmo controller is described in Appendix B information about circuit connection. Figure 4.13 shows the controller mounted on the shoulder of the *WorkPartnre* manipulator. This controller has feedback inputs from potentiometer

Figure 4.12: Elmo controller experimental setup

and encoder. Both communication modes, RS-232 and CAN open, are tested and data collected using composer software developed by Elmo (Elmoc, 2009).

The active current versus mass is measured in two kinds of position configuration. Figure 4.14 is plotted from the data collected in five an identical configuration experiments by taking the average of the data for each load. This figure shows the direct relation between the active current and the load on the end effector. When the load at the end-effector is less than 1.03kg the active current variation is noisy to approximate the end-effector force. For example at the load of 0.43kg the average active current is 0.4325A. However when the load is changed to 0.65kg, the average active current is 0.404A. This variation is shown in Figure 4.14. When the load is greater than 1.3kg, the variation of active current is linear with respect to change in load at the end-effector. If the load is above 1.3kg and difference between two consecutive loads is 0.5kg the load at the end-effector can be estimated using linear approximation, if the position of the manipulator is kept constant.

Figure 4.15 shows the current variation with time when the position varies 400000 ticks of the encoder below the horizontal position and -400000 ticks of

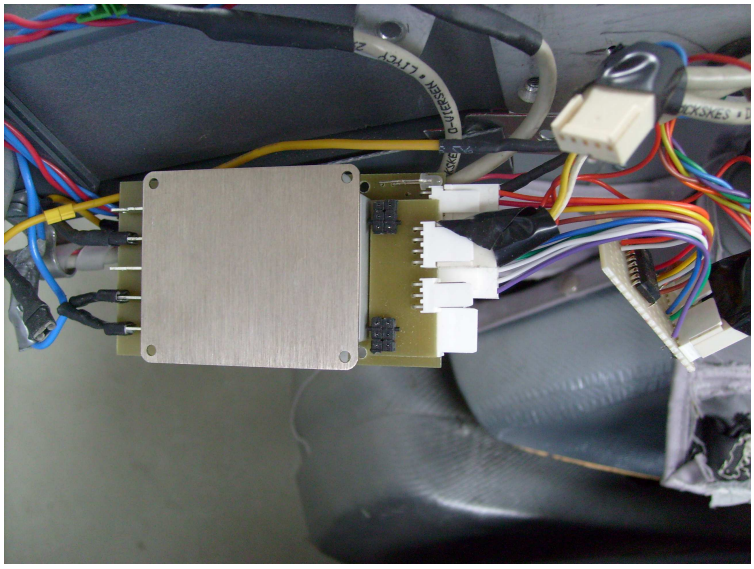

Figure 4.13: Elmo controller mounted at the shoulder of *WorkPartner*

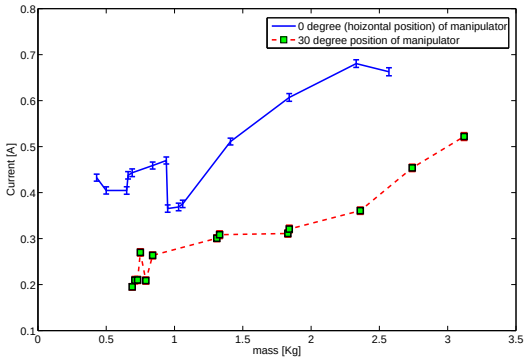

Figure 4.14: Average active current[A] of Elmo controller vs. mass[kg]

the encoder above the horizontal position which is approximately 30 degree to -30 degree above and below the horizontal position of the manipulator. The graph is drawn using the data of five similar experiments based on an identical configuration.

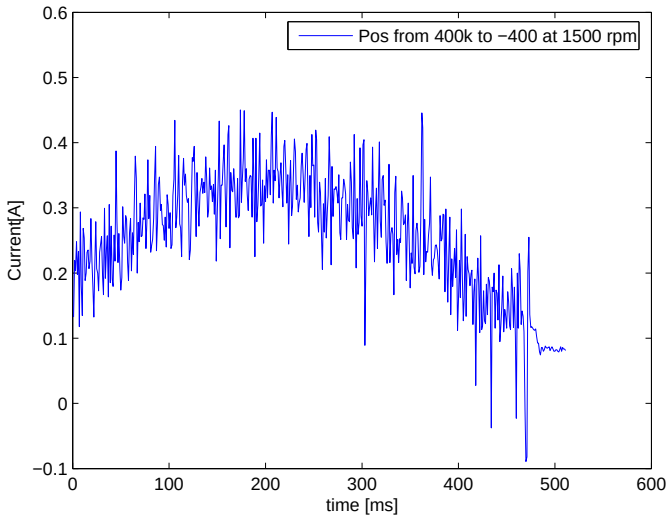

Figure 4.15: Current[A] vs. time[ms] graph when position is changed from 400000 to -400000 ticks at rotational speed of motor 571rpm

4.4.2 Discussion

The results from the experiments show that active current from the Elmo controller has a direct relation to the load on the manipulator as shown in the Figures 4.14 and 4.16. In addition the force has direct relation with the position change as shown in Figures 4.15, 4.18 and 4.19. This data can be used to indicate the direction of manipulator motion and approximate the load at the end-effector, if the applied force at the end-effector of the manipulator is greater than 13N with difference of 5N. This is shown in the experiment done at the horizontal position and 30 degree position from the horizontal. To use Elmo controller for force estimation, mass versus active current model should be done for each possible position of the manipulator. This might be insufficient

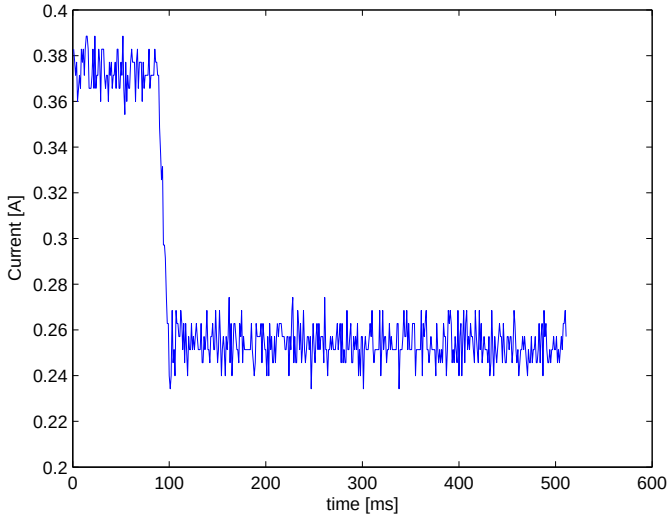

Figure 4.16: The change of current with time when the load is changed from 1.02 to 0 kg keeping the manipulator at horizontal position.

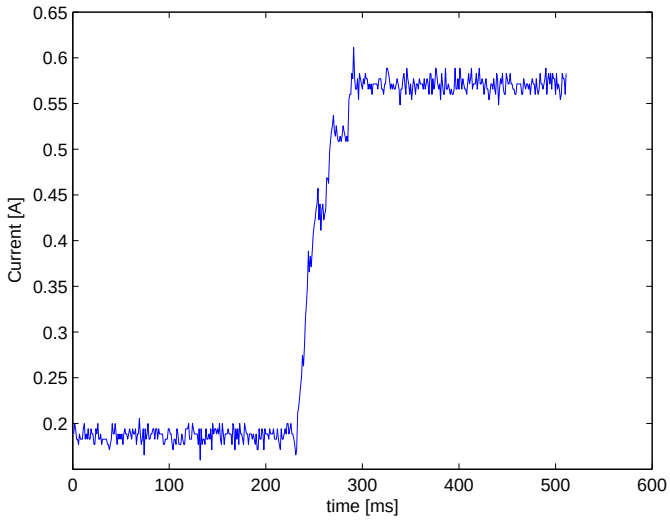

Figure 4.17: The change of current with time when the load is changed from 0 to 2.54 kg keeping the manipulator at horizontal position

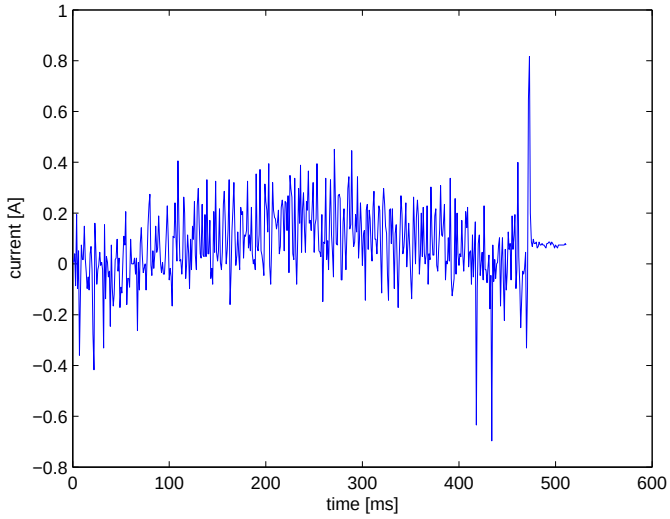

Figure 4.18: Current[A] v time graph using an experiment data when position is changed from 400000 to -400000 ticks at motor rotation speed of 571rpm

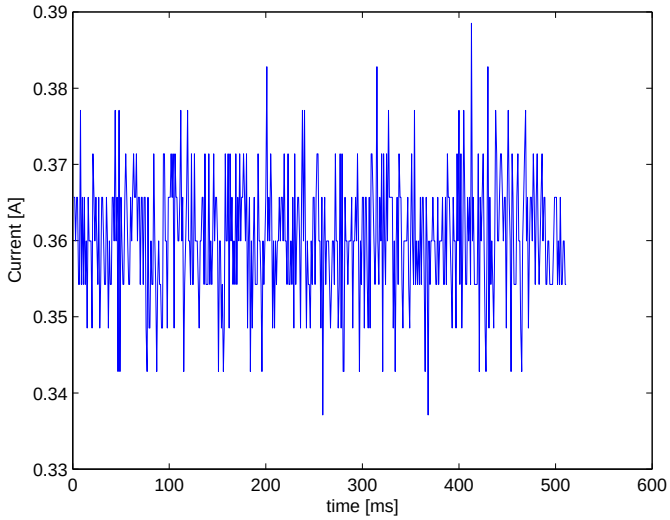

Figure 4.19: An experiment data when the manipulator is without load at horizontal position

for astronaut-robot manipulation assistance since it needs very accurate measurement of the force for safe and efficient interaction. However the controller active current output could be used to approximate force more effectively in applications that require less safety. If filter is added to the controller active current output to decrease the noise level, the approximation of force might be improved. For example Figure 4.19 shows the noise level of the active current at horizontal level with out load.

Chapter 5

Summary and Conclusions

In this thesis manipulator control for PHRI was developed as part of SpacePartner project sponsored by European Space Agency (ESA) and TKK. It addresses problems to enable *WorkPartner* to have safe physical interaction with humans and upgrade the motor controller to support compliance control. Control algorithm was developed using MATLAB and SimPartner robot simulator. Further more it was tested with the real *WorkPartner* manipulator.

The thesis first reviewed the most commonly used control algorithms and force/torque sensors for the application of PHRI. Based on the literature review and *WorkPartner* present condition, control algorithm was chosen as well as required sensor for the manipulator was suggested to implement the control algorithm. Admittance control was chosen because it allows to use the commercial controller with minimum changes in the existing manipulator structure. In addition this control algorithm removes friction effects at the joints using the position controller. This control algorithm also works for both static and moving *WorkPartner*. The control algorithm was implemented on the *WorkPartner* only with stiffness constant since damping on the manipulator was insignificant. The suggested sensor was six-axis force/torque sensor due to accuracy, reliability and repeatability of the sensor compared to others reviewed. However the implementation was done using estimated force from the joint current due to budget constraint.

The *WorkPartner* simulator was modified to suit the implementation of the designed control algorithm. This modification includes adding force sensors, and object that interact with SimPartner which is analogue with human interaction input. The admittance control algorithm was implemented on this simulator. In addition the four chosen demonstration cases were implemented, where the modes corresponded to certain ranges of stiffness values. The results showed stable and safe behavior for the chosen control algorithm for physical interaction purpose, and the possible modes of operations of the manipulator that can be done using this control algorithm.

The two commercial control drivers were mounted at *WorkPartner* joints, tested and analyzed on *WorkPartner* manipulator joints. The active current, i.e. the controlled motor current, from the controller were used to estimate end-effector forces used as a input for the control algorithm on the *WorkPartner*.

This work demonstrated the use of admittance control for follow movement, hold position, adapt movement, and push with force behavior. The control algorithm provides intuitive way to change between the control behaviors. For example *WorkPartner* robot can follow in the direction of the interacting force unlike the industrial robots that follows programmed trajectory even in case of disturbance. This behavior was implemented on *WorkPartner* using a change in active current of the controller, which is interaction force to determine force at the end-effector and by changing the stiffness.

The test results on the simulator and on the real *WorkPartner* showed expected behavior of the chosen modes of operations, which indicated compliance behavior from the implemented control algorithm. The *WorkPartner* simulator had enough performance to develop the algorithm. In contrast the real *WorkPartner* had only few of the required device to develop the algorithm. However the available resource was utilized to test algorithm. The Elmo motor controller was used successfully on a Windows operating system using RS232 and CAN communication interfaces. However, this device had poor accuracy to estimate force from the active current.

The thesis showed that field service robot like *WorkPartner* or industrial robot can include compliance control in addition to planned trajectory following. This makes the robots multipurpose by enabling them to perform tasks in constraint and unconstraint environments. Adding such compliance control is a way to improve the safety when the robots work in unknown environments together with human.

5.1 Future Work

The vision of the Astronaut(human) and robot physical interaction is to have safe manipulation task for natural and seamless execution. Among the potential future works that can be continued on this thesis are test Elmo controller using Linux operating system. In addition, develop human presence and activity recognition algorithm in order to understand human action better as well as implement easier interface such as speech to change mode of operation of the implemented algorithm.

References

Albu-Schaffer, A., Eiberger, O., Grebenstein, M., Haddadin, S., Ott, C., Wimbock, T., Wolf, S., and Hirzinger, G. (2008). *Soft robotics. Robotics & Automation Magazine, IEEE*, 15(3):20–30. ISSN 1070-9932. doi:10.1109/MRA.2008.927979.

Albu-Schaffer, A. and Hirzinger, G. (2002). *Cartesian Impedance Control Techniques for Torque Controlled Light-Weight Robots.*

Bluck and John (2005). *NASA Developing Robots with Human Traits NASA Ames.*
URL: *http://www.nasa.gov/centers/ames/research/exploringtheuniverse/robotshumancoop.html [Accessed on 2009.01.15]*

Buss, S.R. and Kim., J.S. (2004). *Selectively Damped Least Squares for Inverse Kinematics. Journal of Graphics Tools,*, 1:vol. 10, no. 3 37–49. doi:10.1109/ROBOT.2000.844062.

Cabrol, N., Kosmo, J., Trevino, R., and Thomas, H. (1999). *Results of the first astronaut-rover (ASRO) interaction field experiment and recommendations for future planetary surface exploration. Digital Avionics Systems Conference, 1999. Proceedings. 18th*, 2:7.C.3–1–7.C.3–8 vol.2. doi:10.1109/DASC.1999.821996.

Craig, J. (2005). *Introduction to Robotics Mechanics and Control. Pearson Prentice Hall.*

De Santis, A., Siciliano, B., De Luca, A., and Bicchi, A. (2008). *An atlas of physical human-robot interaction. Mechanism and Machine*

Theory, 43(3):253–270. doi:http://dx.doi.org/10.1016/j.mechmachtheory.2007.03.003.
URL: *http://dx.doi.org/10.1016/j.mechmachtheory.2007.03.003*

Diffler, M., Huber, F., Culbert, C., Ambrose, R., and Bluethmann, W. (2003). *Human-robot control strategies for the NASA/DARPA Robonaut. Aerospace Conference, 2003. Proceedings. 2003 IEEE*, 8. ISSN 1095323X.

Elmoc (2009). *Elmo Motion Control.*
URL: *http://www.elmomc.com/*

Fong, T. and Nourbakhsh, I. (2004). *Peer-to-Peer Human-Robot Interaction for Space Exploration.* In *In AIAA Space 2005.*

Heikkilä, S. (2008). *SpacePartner Astronaut and robot cooperation for natural and seamless task execution.* In *Proc. 10th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA).*
URL: *http://automation.tkk.fi/SpacePartner*

Heikkilä, S., Didot, F., and Halme, A. (2008). *Centaur-type service robot technology assessment for astronaut assistant development.* In *Proc. 10th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA).* Noordwijk, Netherlands.

Heiskanen, P., Heikkilä, S., and Halme, A. (2008). *Development of a dynamic mobile robot simulator for astronaut assistance.* In *Proc. 10th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA).* Noordwijk, Netherlands.

Hogan, N. (1987). *Stable execution of contact tasks using impedance control. Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, 4:1047–1054.

Kosuge, K. and Kazamura, N. (1997). *Control of a robot handling an object in cooperation with a human. Robot and Human Communication, 1997. RO-MAN '97. Proceedings., 6th IEEE International Workshop on*, pages 142–147. doi:10.1109/ROMAN.1997.646971.

Lewis, Darren M.Dawson, C.T. (2004). *Robot Manipulator and Control Theory practice.* Marcel Dekker, Inc, second edition, revised and expanded edition.

Morel, G., Malis, E., and Boudet, S. (1998). *Impedance based combination of visual and force control. Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, 2:1743–1748 vol.2. doi: 10.1109/ROBOT.1998.677418.

NASAAmes (2005). *Robonaut Provides Hands-On Assistance in Space.* **URL:** *http://www.cict.nasa.gov/assets/pdf/038CICTISRobonautweb.pdf*

Nelson, B., Morrow, J., and Khosla, P. (1995). *Improved force control through visual servoing. American Control Conference, 1995. Proceedings of the*, 1:380–386 vol.1.

Nguyen, D. and Peters, J. (2008). *Learning Robot Dynamics for Computed Torque Control Using Local Gaussian Processes Regression.* In *LAB-RS '08: Proceedings of the 2008 ECSIS Symposium on Learning and Adaptive Behaviors for Robotic Systems*, pages 59–64. IEEE Computer Society, Washington, DC, USA. ISBN 978-0-7695-3272-1. doi:http://dx.doi.org/10.1109/LAB-RS.2008.16.

of TKK, A.D. (2005). *Work Partner Robot Photo.* **URL:** *http://automation.tkk.fi/WorkPartner/Media*

Okon, D.H.A. and DiCicco, M. (2005). *A Comparison of Force Sensing Techniques for Planetary Manipulation.* Jet Propulsion Laboratory.

Salisbury, J.K. (1980). *Active stiffness control of a manipulator in cartesian coordinates.* In *Decision and Control including the Symposium on Adaptive Processes, 1980 19th IEEE Conference on*, volume 19, pages 95–100. doi:10.1109/CDC.1980.272026.

Semiconductors, N. (2005). *LMD18200 3A 55V H-Bridge.* **URL:** *http://www.national.com/images/pf/LMD18200/01056802.pdf [Accessed on 2009.01.25]*

Siciliano, A.D.S.V.L.B. and Villani, L. (2007). *Human-Robot Interaction Control Using Force and Vision.* In *Springer-Verlag Berlin Heidelberg 2007.*

SOF (2009). *Socity of Robots.* Accessed on 05/02/2009.
URL: *http://www.societyofrobots.com/sensorsforcetorque.shtml*

Sum, E.K. (2008). *SIX-AXIS FORCE/TORQUE SENSOR.* Technical report, University/Industry Liaison Office.

Thomas, P., editor (2005). *Robotics and Automation Handbook.* CRC Press LLC.

Tsumugiwa, T., Yokogawa, R., and Hara, K. (2002). *Variable impedance control with virtual stiffness for human-robot cooperative task (human-robot cooperative peg-in-hole task). SICE 2002. Proceedings of the 41st SICE Annual Conference*, 4:2329–2334 vol.4. doi: 10.1109/SICE.2002.1195769.

Xiao, D., Ghosh, B., Xi, N., and Tarn, T. (2000). *Sensor-based hybrid position/force control of a robot manipulator in an uncalibrated environment. Control Systems Technology, IEEE Transactions on*, 8(4):635–645. ISSN 1063-6536. doi:10.1109/87.852909.

Yoshikawa, T. (2000). *Force control of robot manipulators. Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, 1:220–226 vol.1. doi:10.1109/ROBOT.2000.844062.

Appendix A

Algorithms and Sensors Summary

Algorithm	Stiffness Control	Impedance Control
Working Principle	Based on linear spring relationship	Analogous to Ohm's law
Suitability for PHRI	Good safety and suitable	Safe and suitable
Control variables	position error,contact force	Position and velocity error,contact force
Control Input	Position,force	Position,Velocity,force
Possible Sensors	Encoder,F/T sensor	Encoder,F/T sensor

Table A.1: Summary of Stiffness control and Impedance Control

Algorithm	Computed-Torque Control	Hybrid Position/force control
Working Principle	Model based control	Decouple position , force control
Suitability for PHRI	Poor safety since no force feedback	Good safety but complex
Control variables	Position, Velocity , Acceleration error	Position and force error
Control Input	Desired position, Velocity and Acceleration	Position and force
Possible Sensors	Potentiometer, Encoder	Encoder,F/T sensor

Table A.2: Summary of Computed-Torque Control and Hybrid Position/force control

Sensors	Accuracy	Repeatability	Torque Measuremnt	Force Measurent
Six Axis Force/Torque	Excelent	Excelent	Tx,Ty,Tz	Fx,Fy,Fz
Joint Torque	Very good	Good	Joint Torque	Determine F using Jacobian equation 2.8
Link strain guage	Bad	good	Torque	Determine force using least square estimate
Motor Current	Good	Very good	Torque	Determine F using Jacobian equation 2.8

Table A.3: Summary of Sensors

Appendix B

Workpartner Experimental Manipulator Motor Controller

Overview

Elmo controller module, WHI-5/60, has extensive feedback and control options. The module is shown in the Figure B.1 below, integrated directly to a custom made PCB here in Automation Department. The whole unit is generally smaller than the original controller of the *WorkPartnre*. However, it is slightly ($2mm$) wider than the original.

The WHI-5/60 controller operates from a 12 60VDC supply, and it can drive 5A

Figure B.1: Modified Elmo controller using custom made PCB

continuous current (hence, "5/60"). The power connectors can easily handle this amount of current

Connector pin outs

Five FASTON 4.8mm/0.8mm connectors are used as power and motor connectors. Figure B.2 shows the placement of the connectors. As a difference to the original controller unit, there is now also a protective earth (PE) connector, positioned in the middle. Also, controller's motor phase M1 is not used. Instead, phase M3 has been labeled as "M1".

There are four Molex KK 2.54mm connectors on the other end of the board. These are pin compatible with the ones used on the original *WorkPartner* controller unit. There are some new features however:

- There are inputs for encoder A- and B- signals on previously unused pins of the feedback connector.
- Internally generated +5V is now used as potentiometer supply voltage instead of the external +12V from brake connector.
- There is a 3-pin serial port (RS232) connector for programming the controller.

Figure B.3 shows the positioning of Molex KK 2.54mm connectors.

The pin orders start on the left side of the figure and end on the right side. Pin orders are as follows:

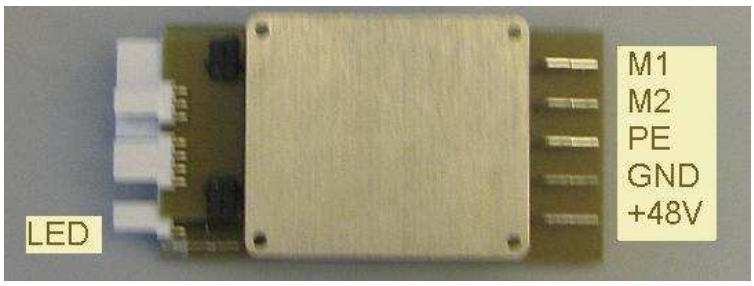

Figure B.2: Positioning of the power and motor connectors

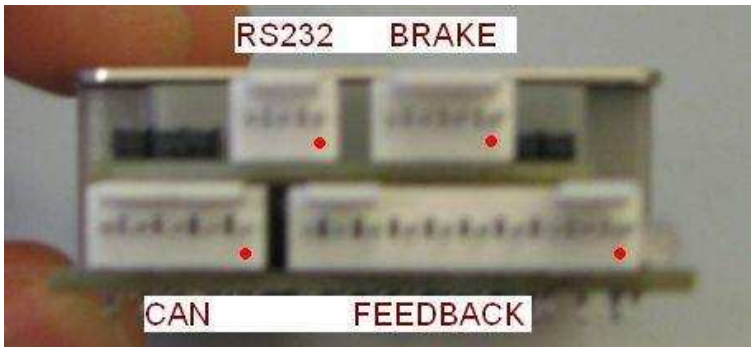

Figure B.3: Positioning of the Molex KK connectors

RS232

1. GND
2. Module TX
3. Module RX

BRAKE

1. BRK1 (directly connected to +12V)
2. BRK2
3. +12V in
4. GND

CAN

1. CANL
2. CANH
3. open
4. open

5. SHIELD

FEEDBACK

1. ENC A+
2. ENC VCC
3. ENC B+
4. open
5. ENC GND
6. POT GND
7. POT OUT
8. ENC A
9. ENC B
10. POT VCC

Appendix C

Denvait-Hartenberg Convention

Any robot can be described kinematically by giving the values of four quantities for each link.

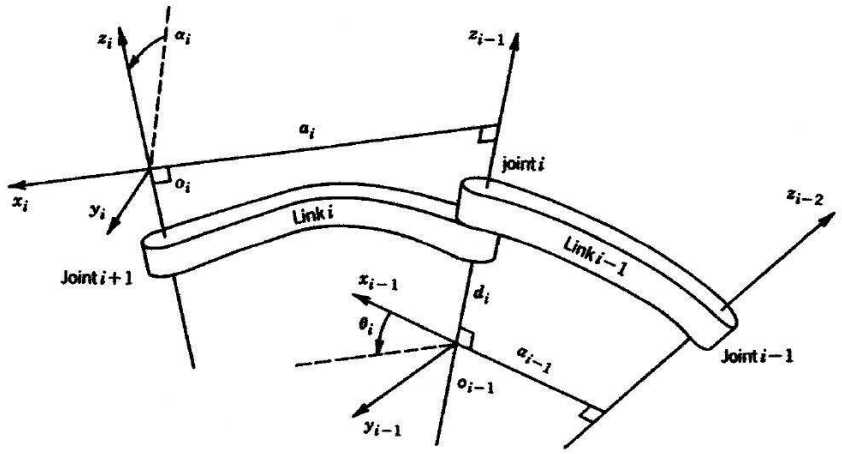

Figure C.1: Denavit-Hartenberg frame assignment (Craig, 2005)

1. Identify the joint axes and imagine infinite lines along them. For steps 2 through 5 below, consider two of these neighboring lines (at axes i and i+1).

2. Identify the common perpendicular between them, or point of intersection id exist. At the point of intersection or the point where the common

perpendicular meets the i^{th} axis, assign the frame origin.

3. Assign axis $\hat{Z}_i$ pointing along the i^{th} joint axis.

4. Assign axis $\hat{X}_i$ pointing along the common perpendicular, or if the axes intersect, assign $\hat{X}_i$ to be normal to the plane containing the two axes.

5. Assign $\hat{Y}_i$ to complete a right-hand coordinate system.

6. Assign frame 0 to match frame 1 when the first joint variables is zero. For n, choose an origin location and $\hat{X}_n$ direction freely, but generally in such way that as many linkage parameters as possible become zero.